

V. G. Ivancevic · T. T. Ivancevic

**Neuro-Fuzzy
Associative Machinery
for Comprehensive
Brain and Cognition
Modelling**

Vladimir G. Ivancevic, Tijana T. Ivancevic

Neuro-Fuzzy Associative Machinery for Comprehensive
Brain and Cognition Modelling

Studies in Computational Intelligence, Volume 45

Editor-in-chief

Prof. Janusz Kacprzyk

Systems Research Institute

Polish Academy of Sciences

ul. Newelska 6

01-447 Warsaw

Poland

E-mail: kacprzyk@ibspan.waw.pl

Further volumes of this series
can be found on our homepage:
springer.com

Vol. 27. Vassilis G. Kaburlasos
*Towards a Unified Modeling and Knowledge-
Representation based on Lattice Theory*, 2006
ISBN 978-3-540-34169-7

Vol. 28. Brahim Chaib-draa, Jörg P. Müller (Eds.)
*Multiagent based Supply Chain
Management*, 2006
ISBN 978-3-540-33875-8

Vol. 29. Sai Sumathi, S.N. Sivanandam
*Introduction to Data Mining and its
Application*, 2006
ISBN 978-3-540-34350-9

Vol. 30. Yukio Ohsawa, Shusaku Tsumoto (Eds.)
Chance Discoveries in Real World Decision Making,
2006
ISBN 978-3-540-34352-3

Vol. 31. Ajith Abraham, Crina Grosan, Vitorino
Ramos (Eds.)
Stigmergic Optimization, 2006
ISBN 978-3-540-34689-0

Vol. 32. Akira Hirose
Complex-Valued Neural Networks, 2006
ISBN 978-3-540-33456-9

Vol. 33. Martin Pelikan, Kumara Sastry, Erick
Cantú-Paz (Eds.)
*Scalable Optimization via Probabilistic
Modeling*, 2006
ISBN 978-3-540-34953-2

Vol. 34. Ajith Abraham, Crina Grosan, Vitorino
Ramos (Eds.)
Swarm Intelligence in Data Mining, 2006
ISBN 978-3-540-34955-6

Vol. 35. Ke Chen, Lipo Wang (Eds.)
Trends in Neural Computation, 2007
ISBN 978-3-540-36121-3

Vol. 36. Ildar Batyrshin, Janusz Kacprzyk, Leonid
Sheremetov, Lotfi A. Zadeh (Eds.)
*Preception-based Data Mining and Decision Making
in Economics and Finance*, 2006
ISBN 978-3-540-36244-9

Vol. 37. Jie Lu, Da Ruan, Guangquan Zhang (Eds.)
E-Service Intelligence, 2007
ISBN 978-3-540-37015-4

Vol. 38. Art Lew, Holger Mauch
Dynamic Programming, 2007
ISBN 978-3-540-37013-0

Vol. 39. Gregory Levitin (Ed.)
*Computational Intelligence in Reliability
Engineering*, 2007
ISBN 978-3-540-37367-4

Vol. 40. Gregory Levitin (Ed.)
*Computational Intelligence in Reliability
Engineering*, 2007
ISBN 978-3-540-37371-1

Vol. 41. Mukesh Khare, S.M. Shiva Nagendra (Eds.)
*Artificial Neural Networks in Vehicular Pollution
Modelling*, 2007
ISBN 978-3-540-37417-6

Vol. 42. Bernd J. Krämer, Wolfgang A. Halang (Eds.)
Contributions to Ubiquitous Computing, 2007
ISBN 978-3-540-44909-6

Vol. 43. Fabrice Guillet, Howard J. Hamilton (Eds.)
Quality Measures in Data Mining, 2007
ISBN 978-3-540-44911-9

Vol. 44. Nadia Nédjah, Luiza de Macedo
Mourelle, Mario Neto Borges, Nival Nunes
de Almeida (Eds.)
Intelligent Educational Machines, 2007
ISBN 978-3-540-44920-1

Vol. 45. Vladimir G. Ivancevic, Tijana T. Ivancevic
*Neuro-Fuzzy Associative Machinery for
Comprehensive Brain and Cognition Modelling*,
2007
ISBN 978-3-540-47463-0

Vladimir G. Ivancevic
Tijana T. Ivancevic

Neuro-Fuzzy Associative Machinery for Comprehensive Brain and Cognition Modelling

With 123 Figures and 4 Tables

 Springer

Dr. Vladimir G. Ivancevic
Human Systems Integration
Land Operations Division Defence
Science & Technology Organisation
PO Box 1500
75 Labs
Edinburgh SA 5111, Australia
E-mail: Vladimir.Ivancevic@dsto.defence.gov.au

Tijana T. Ivancevic
The University of Adelaide
Department of Applied Mathematics
School of Mathematical Sciences
SA 5005, Australia
E-mail: Tijana.Ivancevic@adelaide.edu.au

Library of Congress Control Number: 2006935539

ISSN print edition: 1860-949X

ISSN electronic edition: 1860-9503

ISBN-10 3-540-47463-3 Springer Berlin Heidelberg New York

ISBN-13 978-3-540-47463-0 Springer Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilm or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag. Violations are liable to prosecution under the German Copyright Law.

Springer is a part of Springer Science+Business Media
springer.com

© Springer-Verlag Berlin Heidelberg 2007

The use of general descriptive names, registered names, trademarks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

Cover design: deblik, Berlin

Typesetting by the authors and SPi

Printed on acid-free paper SPIN: 11898467 89/SPi 5 4 3 2 1 0

Preface

Neuro-Fuzzy Associative Machinery for Comprehensive Brain and Cognition Modeling is a graduate-level monographic textbook. It represents a comprehensive introduction into both conceptual and rigorous brain and cognition modeling. It is devoted to understanding, prediction and control of the fundamental mechanisms of brain functioning. The objective of the monograph is to provide a reader with a scientific tool that will enable him/her to actually *perform* a competitive research in brain and cognition modeling.

This book has four Chapters and Appendix. The first Chapter gives an introduction into brain anatomy, physiology and biophysics, as well as psychology and engineering of cognition. It also introduces the functorial meta-language, which is essential for global comprehension of brain and cognition. The second Chapter reviews various approaches (or, functors) currently used for brain and cognition modeling, including: (i) nonlinear dynamics and chaos, (ii) control and complex systems and networks, (iii) intelligent (BDI) agents, (iv) neural networks, (v) cellular automata and genetic algorithms, (vi) fuzzy logic, and (vii) synergetics. The third Chapter develops the neurodynamics, neuro-synergetics and synchronization approach of brain and cognition modeling. The fourth Chapter further develops fuzzy and neuro-fuzzy systems. It also gives our neuro-fuzzy modeling approach to real-life games. The Appendix has two sections: (i) universal language of categories and functors, and (ii) glossary of frequent neurophysiological terms.

Target readership includes all researchers and students interested in brain and cognition modeling (in computer science, engineering, mathematics, physics, chemistry, biology, psychology, sociology, economics, medicine, etc.), working both in industry (i.e., clinics) and academia.

Adelaide,
Aug 2006

*V. Ivancevic, Defence Science & Technology Organisation,
Australia, e-mail: Vladimir.Ivancevic@dsto.defence.gov.au*

*T. Ivancevic, School of Mathematics, The University of Adelaide,
e-mail: Tijana.Ivancevic@adelaide.edu.au*

Dedicated to Nitya, Atma and Kali

Acknowledgments

The authors wish to thank Land Operations Division, Defence Science & Technology Organisation, Australia, for the support in developing the *Human Biodynamics Engine* (HBE) and all the HBE-related text in this monograph.

We also express our gratitude to *Springer* book series *Studies in Computational Intelligence* and especially to the Editor, Professor Janusz Kacprzyk.

Contents

0.1	Glossary of Frequently Used Symbols	1
1	Introduction to Brain and Cognition	3
1.1	Brain	3
1.1.1	Basics of Brain Physiology	10
1.1.2	Modern 3D Brain Imaging	30
1.2	Cognition and Associative Memory	37
1.3	Categorical Language for Neuro-Cognition.....	43
1.3.1	Categorical Patterns and Hierarchical Links	44
1.3.2	A General Natural System.....	46
1.3.3	The Category of Neurons	47
1.3.4	Memory Evolutive System	47
1.3.5	Neural System in a Category	49
1.4	Hierarchical Associative Machinery of n -Functors	51
1.4.1	Preliminary Examples of Brain and Cognition Functors	56
2	Various Approaches (Functors) to Brain and Cognition	
	Modeling	61
2.1	Nonlinear Dynamics and Chaos Approach	61
2.1.1	Chaos Theory Basics	62
2.1.2	Chaos Control	81
2.1.3	Fractal Calculus and Chaos Control	103
2.1.4	Josephson Junctions and Quantum Computing	106
2.1.5	Solitons	110
2.1.6	Chaos Field Theory.....	116
2.1.7	Geometrical Nonlinear Dynamics	118
2.1.8	Dynamics of n -Functors	120
2.2	Systems and Networks Approach	122
2.2.1	MIMO Systems and Kalman Filtering.....	122
2.2.2	Nonlinear Control Systems	128
2.2.3	Complex Adaptive Systems	144

2.2.4	Complex Networks and Synchronization Dynamics	147
2.2.5	Small-Worlds and Social Networks	152
2.3	Intelligent (BDI) Agents Approach	155
2.3.1	Agents Environments	161
2.3.2	Agents' Reasoning and Learning	164
2.4	Neural-Networks Approach	183
2.4.1	Biological Versus Artificial Neural Nets	184
2.4.2	Common Discrete ANNs	185
2.4.3	Common Continuous ANNs	209
2.4.4	Support Vector Machines	217
2.4.5	Dynamical Nets for Thought Process Modeling	220
2.5	Cellular and Evolutionary Approach	228
2.5.1	Cellular Automata	229
2.5.2	Evolutionary Algorithms	237
2.6	Fuzzy-Logic Approach	246
2.6.1	The Concept of Fuzziness	246
2.6.2	Fuzzy Inference Engine	255
2.6.3	Fuzzy Logic Control	258
2.7	Synergetics Approach	267
2.7.1	Phase Transitions	270
2.7.2	Hamiltonian Biodynamics	271
2.7.3	Kick Dynamics and Dissipation-Fluctuation Theorem	273
2.7.4	Mezoscopic Derivation of Order Parameters	276
2.7.5	Example: Synergetic Control of Biodynamics	278
2.7.6	Example: Chaotic Psychodynamics of Perception	280
3	Neuro-Dynamics, Synergetics and Synchronization	285
3.1	Distinguished ANN Models	285
3.1.1	Generalized Kohonen's SOM	285
3.1.2	Dynamics of Hopfield's Associative Recurrent Nets	289
3.1.3	A Self-Organizing Bidirectional Competitive Net	296
3.1.4	Vector AM-Neural Nets	297
3.1.5	Small-World AM-Neural Nets	307
3.2	Local Neurodynamics	311
3.2.1	Microscopic Neurodynamics and Quantum Brain	311
3.2.2	Macroscopic Neurodynamics	324
3.2.3	Darwinian Oscillatory Neural Net	345
3.2.4	Oscillatory Phase Synchronization in Neurodynamics	347
3.2.5	Synchronization in Cellular Automata	355
3.2.6	Kuramoto-Type Dynamics of Complex Nets	360
3.2.7	Oscillator AM with Scattered Natural Frequencies	363
3.2.8	AM Neural Nets with Self-Control	367
3.2.9	BAM and Statistical Neurodynamics	375
3.2.10	Firing Correlation in Layered AM Neural Nets	384
3.3	Synergetics of Recurrent and Attractor Neural Nets	388

3.3.1	Stochastic Dynamics of Neuronal Firing States	389
3.3.2	Synaptic Symmetry and Lyapunov Functions	394
3.3.3	Balance and Equilibrium Statistical Mechanics	396
3.3.4	Simple Recurrent Nets with Binary Neurons	402
3.3.5	Simple Recurrent Nets of Coupled Oscillators	410
3.3.6	Attractor Neural Nets with Binary Neurons	418
3.3.7	Attractor Neural Nets with Continuous Neurons	430
3.3.8	Correlation- and Response-Functions	436
3.4	Synchronization in Arrays of Josephson Junctions	446
3.4.1	Phase Model for Underdamped JJL	449
3.4.2	Comparison of LKM2 and RCSJ Models	454
3.4.3	'Small-World' Connections in JJL Arrays	455
3.5	Global Neurodynamics	458
3.5.1	A Path-Integral AM-Model	458
3.5.2	General Path-Integral Approach to Attractor NNs	467
3.5.3	Hierarchical Self-Programming in Neural Nets	489
3.5.4	Neural Path-Integral Model for the Cerebellum	495
3.5.5	Brain-Like Control Functor in Biomechanics	501
3.5.6	Concurrent and Weak Functorial Machines	513
3.5.7	Brain-Mind Functorial Machines	525
3.5.8	Path-Integral Psychodynamics	532
4	Fuzzy Systems	547
4.1	Fuzzy-Logic Control	547
4.2	Fuzzy Cognitive Maps	552
4.2.1	FCM Definitions	553
4.2.2	FCM Properties and Models	556
4.3	Takagi-Sugeno Fuzzy Dynamics	559
4.3.1	The Basic Takagi-Sugeno Model	559
4.3.2	Local and Global Fuzzy Dynamical Models	560
4.3.3	Affine Fuzzy System	561
4.3.4	Fuzzy Modeling and Control of Chaotic Systems	562
4.4	Evolutionary Fuzzy Recurrent Control and Computational Geometry	565
4.4.1	Fuzzy Recurrent Voronoi Model	567
4.4.2	RFV Design with Evolutionary Algorithms	570
4.4.3	Fuzzy-Genetic Properties of RFV	571
4.4.4	Application to Evolutionary Robotics	572
4.5	Fuzzy Discrete Event Systems	574
4.5.1	Crisp Discrete Event Systems	576
4.5.2	Fuzzy Automaton Model	577
4.5.3	Controllability of Fuzzy Languages	581
4.6	Neuro-Fuzzy Associative n -Functors	594
4.6.1	High-Resolution Functorial Agents	594
4.6.2	Application to Real-Life Games	596

XII Contents

4.6.3	Generic Functorial FAM–Agents.....	600
5	Appendix	601
5.1	Universal Associative Language of Categories and Functors.....	601
5.1.1	Maps	601
5.1.2	Categories	613
5.1.3	Functors	616
5.1.4	Natural Transformations	618
5.1.5	Limits and Colimits	621
5.1.6	Adjunction	622
5.1.7	n –Categories and n –Functors	623
5.2	Glossary of Frequent Neurophysiological Terms.....	630
	References	651
	Index	719

0.1 Glossary of Frequently Used Symbols

General

- ‘iff’ means ‘if and only if’;
- ‘r.h.s’ means ‘right hand side’; ‘l.h.s’ means ‘left hand side’;
- *Einstein’s summation convention over repeated indices* (not necessarily one up and one down) *is assumed in the whole text*, unless explicitly stated otherwise. (Note that this is not a custom among the connectionists; however, it is a custom among the dynamicists.)

Sets

$\mathbb{N}$ – natural numbers;
 $\mathbb{Z}$ – integers;
 $\mathbb{R}$ – real numbers;
 $\mathbb{C}$ – complex numbers.

Maps

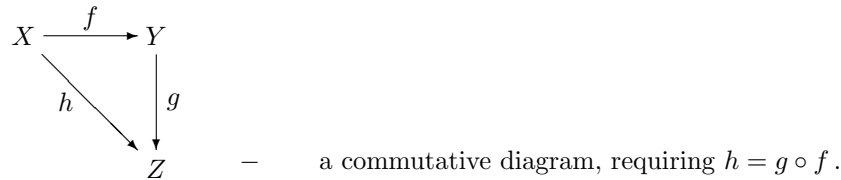
$f : A \rightarrow B$ – a function, (or map) between sets $A \equiv \text{Dom } f$ and $B \equiv \text{Cod } f$;

$\text{Ker } f = f^{-1}(e_B)$ – a kernel of f ;

$\text{Im } f = f(A)$ – an image of f ;

$\text{Coker } f = \text{Cod } f / \text{Im } f$ – a cokernel of f ;

$\text{Coim } f = \text{Dom } f / \text{Ker } f$ – a coimage of f ;



Derivatives and Integrals

$C^k(A, B)$ – set of k –times differentiable functions between sets A to B ;

$C^\infty(A, B)$ – set of *smooth* functions between sets A to B ;

$C^0(A, B)$ – set of *continuous* functions between sets A to B ;

$f'(x) = \frac{df(x)}{dx}$ – derivative of f with respect to x ;

$\dot{x}$ – total time derivative of x ;

$\partial_t \equiv \frac{\partial}{\partial t}$ – partial time derivative;

$\partial_{x^i} \equiv \partial_i \equiv \frac{\partial}{\partial x^i}$ – partial coordinate derivative;

$\dot{f} = \partial_t f + \partial_{x^i} f \dot{x}^i$ – total time derivative of the scalar field $f = f(t, x^i)$;

$u_t \equiv \partial_t u$, $u_x \equiv \partial_x u$, $u_{xx} \equiv \partial_{x^2} u$ – only in partial differential equations;
 $L_{x^i} \equiv \partial_{x^i} L$, $L_{\dot{x}^i} \equiv \partial_{\dot{x}^i} L$ – coordinate and velocity partial derivatives of the Lagrangian function;
 $\nabla = \nabla(g)$ – affine Levi–Civita connection on a smooth manifold M with Riemannian metric tensor $g = g_{ij}$;
 Γ_{jk}^i – Christoffel symbols of the affine connection ∇ ;
 $\nabla_X T$ – covariant derivative of the tensor–field T with respect to the vector–field X , defined by means of Γ_{jk}^i ;
 $\mathcal{L}_X T$ – Lie derivative of the tensor–field T in direction of the vector–field X ;
 $[X, Y]$ – Lie bracket (commutator) of two vector–fields X and Y ;
 $[F, G]$ – Poisson bracket of two functions F and G ;
 $\{F, G\}$ – Lie–Poisson bracket of two functions F and G ;
 $\oint$ – Feynman path symbol, denoting integration over continuous spectrum of smooth paths and summation over discrete spectrum of Markov chains; e.g., $\oint \mathcal{D}[x] e^{iS[x]}$ denotes the path integral (i.e., sum–over–histories) over all possible paths $x^i = x^i(t)$ defined by the Hamilton action, $S[x] = \frac{1}{2} \int_{t_0}^{t_1} g_{ij} \dot{x}^i \dot{x}^j dt$, while $\oint \mathcal{D}[\Phi] e^{iS[\Phi]}$ denotes the path integral over all possible fields $\Phi^i = \Phi^i(x)$ defined by some field action $S[\Phi]$.

Introduction to Brain and Cognition

In this Chapter we give introduction to brain and cognition. Material presented here is the necessary background for comprehensive reading of the subsequent Chapters.

1.1 Brain

Recall that *human brain* is the most complicated mechanism in the Universe. Roughly, it has its own physics and physiology. A closer look at the brain points to a rather *recursively hierarchical structure* and a very elaborate organization (see [II06a] for details). An average brain weighs about 1.3 kg, and it is made of: $\sim 77\%$ water, $\sim 10\%$ protein, $\sim 10\%$ fat, $\sim 1\%$ carbohydrates, $\sim 0.01\%$ DNA/RNA. The largest part of the human brain, the cerebrum, is found on the top and is divided down the middle into left and right cerebral hemispheres, and front and back into *frontal lobe*, *parietal lobe*, *temporal lobe*, and *occipital lobe*. Further down, and at the back lies a rather smaller, spherical portion of the brain, the *cerebellum*, and deep inside lie a number of complicated structures like the *thalamus*, *hypothalamus*, *hippocampus*, etc.

Both the cerebrum and the cerebellum have comparatively thin outer surface layers of grey matter and larger inner regions of white matter. The grey regions constitute what is known as the *cerebral cortex* and the *cerebellar cortex*. It is in the *grey matter* where various kinds of *computational tasks* seem to be performed, while the *white matter* consists of long nerve fibers (axons) carrying signals from one part of the brain to another. However, despite of its amazing computational abilities, brain is not a computer, at least not a ‘Von Neumann computer’ [Neu58], but rather a huge, hierarchical, neural network. It is the cerebral cortex that is central to the higher brain functions, speech, thought, complex movement patterns, etc. On the other hand, the cerebellum seems to be more of an ‘automaton’. It has to do more with precise coordination and control of the body, and with skills that have become our ‘second nature’. Cerebellum actions seem almost to take place by themselves, without

thinking about them. They are very similar to the unconscious reflex actions, e.g., reaction to pinching, which may not be mediated by the brain, but by the upper part of the spinal column.

Various regions of the cerebral cortex are associated with very specific functions. The *visual cortex*, a region in the occipital lobe at the back of the brain, is responsible for the reception and interpretation of vision. The *auditory cortex*, in the temporal lobe, deals mainly with analysis of sound, while the *olfactory cortex*, in the frontal lobe, deals with smell. The *somatosensory cortex*, just *behind* the division between frontal and parietal lobes, has to do with the sensations of touch. There is a very specific mapping between the various parts of the surface of the body and the regions of the somatosensory cortex. In addition, just in *front* of the division between the frontal and parietal lobes, in the frontal lobe, there is the *motor cortex*. The *motor cortex* activates the movement of different parts of the body and, again here, there is a very specific mapping between the various muscles of the body and the regions of the motor cortex. All the above mentioned regions of the cerebral cortex are referred to as *primary*, since they are the one most directly concerned with the input and output of the brain. Near to these primary regions are the *secondary* sensory regions of the cerebral cortex, where information is processed, while in the *secondary motor* regions, conceived plans of motion get translated into specific directions for actual muscle movement by the primary motor cortex. The most abstract and sophisticated activity of the brain is carried out in the remaining regions of the cerebral cortex, the *association cortex*.

The basic building blocks of the brain are the nerve cells or neurons. Among about 200 types of different basic types of human cells, the neuron is one of the most specialized, exotic and remarkably versatile cell. The neuron is highly unusual in three respects: its *variation in shape*, its *electrochemical function*, and its *connectivity*, i.e., its ability to link up with other neurons in networks. Let us start with a few elements of neuron microanatomy [KSJ91, BS91]. There is a central starlike bulb, called the *soma*, which contains the nucleus of the cell. A long nerve fibre, known as the *axon*, stretches out from one end of the soma. Its length, in humans, can reach up to few *cm*. The function of an axon is to transmit the neuron's output signal, in which it acts like a *coaxial cable*. The axon has the ability of multiple bifurcation, branching out into many smaller nerve fibers, and the very end of which there is always a *synaptic knob*. At the other end of the soma and often springing off in all directions from it, are the tree-like *dendrites*, along which input data are carried into the soma. The whole nerve cell, as basic unit, has a cell membrane surrounding soma, axon, synoptic knobs, and dendrites. Signals pass from one neuron to another at junctions known as *synapses*, where a synaptic knob of one neuron is attached to another neuron's soma or dendrites. There is very narrow gap, of a few *nm*, between the synaptic knob and the soma/dendrite to where the *synaptic cleft* is attached. The signal from one neuron to another has to propagate across this gap.

A nerve fibre is a cylindrical tube containing a mixed solution of NaCl and KCl, mainly the second, so there are Na^+ , K^+ , and Cl^- ions within the tube. Outside the tube the same type of ions are present but with more Na^+ than K^+ . In the *resting* state there is an excess of Cl^- over Na^+ and K^+ inside the tube, giving it a negative charge, while it has positive charge outside. A nerve signal is a region of *charge reversal* traveling along the fibre. At its head, *sodium gates* open to allow the sodium to flow inwards and at its tail *potassium gates* open to allow potassium to flow outwards. Then, metabolic pumps act to restore order and establish the *resting state*, preparing the nerve fibre for another signal. There is no major material (ion) transport that produces the signal, just in and out local movements of ions, across the cell membranes, i.e., a *small* and *local* depolarization of the cell. Eventually, the nerve signal reaches the attached synaptic knob, at the very end of the nerve fibre, and triggers it to emit chemical substances, known as *neurotransmitters*. It is these substances that travel across the synaptic cleft to another neuron's soma or dendrite. It should be stressed that the signal here is not electrical, but a chemical one. What really is happening is that when the nerve signal reaches the synaptic knob, the local depolarization cause little bags immersed in the *vesicular grid*, the *vesicles* containing molecules of the neurotransmitter chemical (e.g., acetylcholine) to release their contents from the neuron into the synaptic cleft, the phenomenon of *exocytosis*. These molecules then diffuse across the cleft to interact with *receptor proteins* on receiving neurons. On receiving a neurotransmitter molecule, the receptor protein opens a gate that causes a local depolarization of the receiver neuron.

It depends on the nature of the synaptic knob and of the specific synaptic junction, if the next neuron would be encouraged to *fire*, i.e., to start a new signal along its own axon, or it would be discouraged to do so. In the former case we are talking about *excitatory synapses*, while in the latter case about *inhibitory synapses*. At any given moment, one has to add up the effect of all excitatory synapses and subtract the effect of all the inhibitory ones. If the net effect corresponds to a positive electrical potential difference between the inside and the outside of the neuron under consideration, *and* if it is bigger than a critical value, then the neuron *fires*, otherwise it stays mute.

The basic dynamical process of *neural communication* can be summarized in the following three steps: [Nan95]

1. The neural axon is an *all or none* state. In the *all state* a signal, called a *spike* or *action potential* (AP), propagates indicating that the summation performed in the soma produced an amplitude of the order of tens of mV. In the *none state* there is no signal traveling in the axon, only the resting potential ($\sim -70\text{mV}$). It is essential to notice that the presence of a traveling signal in the axon, *blocks* the possibility of transmission of a second signal.
2. The nerve signal, upon arriving at the ending of the axon, triggers the emission of neurotransmitters in the synaptic cleft, which in turn cause

the receptors to open up and allow the penetration of ionic current into the *post synaptic* neuron. The *efficacy* of the synapse is a parameter specified by the amount of penetrating current per presynaptic spike.

3. The post synaptic potential (PSP) diffuses toward the soma, where all inputs in a short period, from all the presynaptic neurons connected to the postsynaptic are summed up. The amplitude of individual PSP's is about 1mV, thus quite a number of inputs is required to reach the 'firing' threshold, of tens of mV. Otherwise the postsynaptic neuron remains in the *resting* or *none* state.

The cycle-time of a neuron, i.e., the time from the emission of a spike in the presynaptic neuron to the emission of a spike in the postsynaptic neuron is of the order of 1 – 2 ms. There is also some recovery time for the neuron, after it fired, of about 1 – 2 ms, independently of how large the amplitude of the depolarizing potential would be. This period is called the absolute refractory period of the neuron. Clearly, it sets an upper bound on the spike frequency of 500 – 1000/sec. In the types of neurons that we will be interested in, the spike frequency is considerably lower than the above upper bound, typically in the range of 100/sec, or even smaller in some areas, at about 50/sec. It should be noticed that this rather exotic neural communication mechanism works very efficiently and it is employed universally, both by vertebrates and invertebrates. The vertebrates have gone even further in perfection, by protecting their nerve fibers by an insulating coating of myelin, a white fatty substance, which incidentally gives the white matter of the brain, discussed above, its color. Because of this insulation, the nerve signals may travel undisturbed at about 120 m/sec [Nan95]

A very important anatomical fact is that each neuron receives some 10^4 synaptic inputs from the axons of other neurons, usually one input per presynaptic neuron, and that each branching neural axon forms about the same number ($\sim 10^4$) of synaptic contacts on other, postsynaptic neurons. A closer look at our cortex then would expose a mosaic-type structure of assemblies of a few thousand densely connected neurons. These assemblies are taken to be the basic cortical processing *modules*, and their size is about $1(\text{mm})^2$. The neural connectivity gets much sparser as we move to larger scales and with much less feedback, allowing thus for autonomous local collective, parallel processing and more serial and integrative processing of local collective outcomes. Taking into account that there are about 10^{11} nerve cells in the brain (about 7×10^{10} in the cerebrum and 3×10^{10} in the cerebellum), we are talking about 10^{15} synapses.

While the dynamical process of neural communication suggests that the brain action looks a lot like a computer action, there are some fundamental differences having to do with a basic brain property called *brain plasticity*. The interconnections between neurons are not fixed, as is the case in a computer-like model, but are changing all the time. These are *synaptic junctions* where the communication between different neurons actually takes place. The synap-

tic junctions occur at places where there are *dendritic spines* of suitable form such that contact with the synaptic knobs can be made. Under certain conditions these dendritic spines can shrink away and break contact, or they can grow and make new contact, thus determining the *efficacy* of the synaptic junction. Actually, it seems that it is through these dendritic spine changes, in synaptic connections, that long-term memories are laid down, by providing the means of storing the necessary information. A supporting indication of such a conjecture is the fact that such dendritic spine changes occur within *seconds*, which is also how long it takes for permanent memories to be laid down [Pen89, Pen94, Pen97, Sta93].

Furthermore, a very useful set of phenomenological rules has been put forward by Hebb [Heb49], the *Hebb rules*, concerning the underlying mechanism of brain plasticity. According to Hebb, a synapse between neuron 1 and neuron 2 would be strengthened whenever the firing of neuron 1 is followed by the firing of neuron 2, and weakened whenever it is not. It seems that *brain plasticity* is a *fundamental property* of the activity of the brain.

Many mathematical models have been proposed to try to simulate *learning process*, based upon the close resemblance of the dynamics of neural communication to computers and implementing, one way or another, the essence of the Hebb rules. These models are known as *neural networks*. They are closely related to *adaptive Kalman filtering* (see (2.2.1) above) as well as *adaptive control* (see [II06b]).

Let us try to construct a *neural network* model for a set of N interconnected neurons (see e.g., [Ama77]). The activity of the neurons is usually parameterized by N functions $\sigma_i(t)$, $i = 1, 2, \dots, N$, and the synaptic strength, representing the synaptic efficacy, by $N \times N$ functions $j_{ik}(t)$. The total stimulus of the network on a given neuron (i) is assumed to be given simply by the sum of the stimuli coming from each neuron

$$S_i(t) = j_{ik}(t)\sigma_k(t), \quad (\text{summation over } k)$$

where we have identified the individual stimuli with the product of the synaptic strength j_{ik} with the activity σ_k of the neuron producing the individual stimulus. The dynamic equations for the neuron are supposed to be, in the simplest case

$$\dot{\sigma}_i = F(\sigma_i, S_i), \quad (1.1)$$

with F a nonlinear function of its arguments. The dynamic equations controlling the time evolution of the synaptic strengths $j_{ik}(t)$ are much more involved and only partially understood, and usually it is assumed that the j -dynamics is such that it produces the synaptic couplings. The simplest version of a neural network model is the *Hopfield model* [Hop82]. In this model the neuron activities are conveniently and conventionally taken to be *switch*-like, namely ± 1 , and the time t is also an integer-valued quantity. This *all(+1)* or *none(-1)* neural activity σ_i is based on the neurophysiology. The

choice ± 1 is more natural the usual ‘binary’ one ($b_i = 1$ or 0), from a physicist’s point of view corresponding to a two-state system, like the fundamental elements of the ferromagnet, i.e., the electrons with their spins up (+) or (−).

The increase of time t by one unit corresponds to one step for the dynamics of the neuron activities obtainable by applying (for all i) the rule

$$\sigma_i(t + \frac{i+1}{N}) = \text{sign}(S_i(t + i/N)), \quad (1.2)$$

which provides a rather explicit form for (1.1). If, as suggested by the Hebb rules, the j matrix is *symmetric* ($j_{ik} = j_{ki}$), the Hopfield dynamics [Hop82] corresponds to a sequential algorithm for looking for the minimum of the Hamiltonian

$$H = -S_i(t)\sigma_i(t) = -j_{ik}\sigma_i(t)\sigma_k(t).$$

The Hopfield model, at this stage, is very similar to the dynamics of a statistical mechanics *Ising-type* [Gol92], or more generally a *spin-glass*, model [Ste92]. This *mapping* of the Hopfield model to a spin-glass model is highly advantageous because we have now a justification for using the statistical mechanics language of phase transitions, like critical points or attractors, etc, to describe neural dynamics and thus brain dynamics. This simplified Hopfield model has many *attractors*, corresponding to many different *equilibrium* or *ordered* states, endemic in spin-glass models, and an unavoidable prerequisite for successful storage, in the brain, of many different patterns of activities. In the neural network framework, it is believed that an internal representation (i.e., a pattern of neural activities) is associated with each object or category that we are capable of recognizing and remembering. According to neurophysiology, it is also believed that an object is memorized by suitably changing the synaptic strengths. The so-called *associative memory* (see next section) is generated produced in this scheme as follows [Nan95]: An external stimulus, suitably involved, produces synaptic strengths such that a specific learned pattern $\sigma_i(0) = P_i$ is ‘printed’ in such a way that the neuron activities $\sigma_i(t) \sim P_i$ (II *learning*), meaning that the σ_i will remain for all times close to P_i , corresponding to a stable attractor point (III *coded brain*). Furthermore, if a *replication signal* is applied, pushing the neurons to σ_i values *partially* different from P_i , the neurons should evolve toward the P_i . In other words, the memory is able to retrieve the information on the whole object, from the knowledge of a part of it, or even in the presence of wrong information (IV *recall process*). Clearly, if the external stimulus is very different from any pre-existing $\sigma_i = P_i$ pattern, it may either create a new pattern, i.e., create a new attractor point, or it may reach a chaotic, random behavior (I *uncoded brain*).

Despite the remarkable progress that has been made during the last few years in understanding brain function using the neural network paradigm, it is fair to say that neural networks are rather artificial and a very long way from providing a realistic model of brain function. It seems likely that the mechanisms controlling the changes in synaptic connections are much

more complicated and involved than the ones considered in NN, as utilizing cytoskeletal restructuring of the sub-synaptic regions. *Brain plasticity* seems to play an essential, central role in the workings of the brain! Furthermore, the ‘binding problem, i.e., how to *bind* together all the neurons firing to different features of the same object or category, especially when more than one object is perceived during a *single* conscious perceptual moment, seems to remain unanswered [Nan95]. In this way, we have come a long way since the times of the ‘grandmother neuron’, where a *single* brain location was invoked for self observation and control, identified with the pineal glands by Descartes [Ami89].

It has been long suggested that different groups of neurons, responding to a common object/category, fire *synchronously*, implying *temporal correlations* [SG95]. If true, such correlated firing of neurons may help us in resolving the binding problem [Cri94]. Actually, brain waves recorded from the scalp, i.e., the EEGs, suggest the existence of some sort of *rhythms*, e.g., the ‘ α -rhythms’ of a frequency of 10 Hz. More recently, oscillations were clearly observed in the visual cortex. Rapid oscillations, above EEG frequencies in the range of 35 to 75 Hz, called the ‘ γ -oscillations’ or the ‘40 Hz oscillations’, have been detected in the cat’s visual cortex [SG95]. Furthermore, it has been shown that these oscillatory responses can become *synchronized* in a stimulus-dependent manner. Studies of auditory-evoked responses in humans have shown inhibition of the 40 Hz coherence with *loss of consciousness* due to the induction of general anesthesia [SB74]. These striking results have prompted Crick and Koch to suggest that this *synchronized firing* on, or near, the beat of a ‘ γ -oscillation’ (in the 35–75 Hz range) might be the *neural correlate* of *visual awareness* [Cri94]. Such a behavior would be, of course, a very special case of a much more general framework where coherent firing of *widely-distributed, non-local* groups of neurons, in the ‘beats’ of x -oscillation (of specific frequency ranges), *bind* them together in a mental representation, expressing the *oneness* of *consciousness* or *unitary sense of self*. While this is a bold suggestion [Cri94], it should be stressed that in a physicist’s language it corresponds to a phenomenological explanation, not providing the underlying physical mechanism, based on neuron dynamics, that triggers the synchronized neuron firing [Nan95]. On the other hand, the *Crick–Koch binding hypothesis* [CK90, Cri94] is very suggestive (see Figure 1.1) and in compliance with the *central biodynamic adjunction* [II06a] (explained later)

$$\text{coordination} = \text{sensory} \dashv \text{motor} : \text{brain} \rightleftarrows \text{body}.$$

On the other hand, E.M. Izhikevich, Editor-in-Chief of the new Encyclopedia of Computational Neuroscience, considers brain as a *weakly-connected neural network* [Izh99b], consisting of n quasi-periodic cortical oscillators $X_1, \dots, X_n$ forced by the thalamic input X_0 (see Figure 1.2)

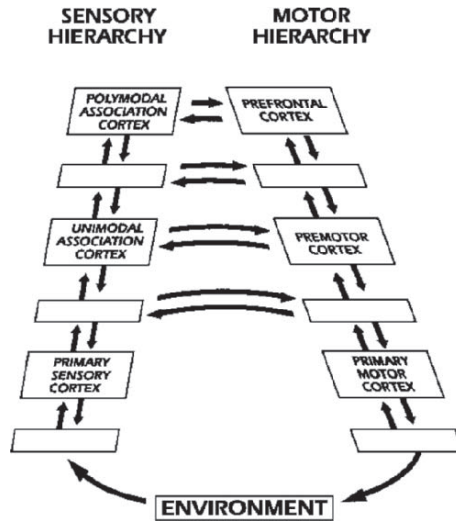


Fig. 1.1. Fiber connections between cortical regions participating in the *perception-action cycle*, reflecting again our sensory-motor adjunction. Empty rhomboids stand for intermediate areas or subareas of the labeled regions. Notice that there are connections between the two hierarchies at several levels, not just at the top level.

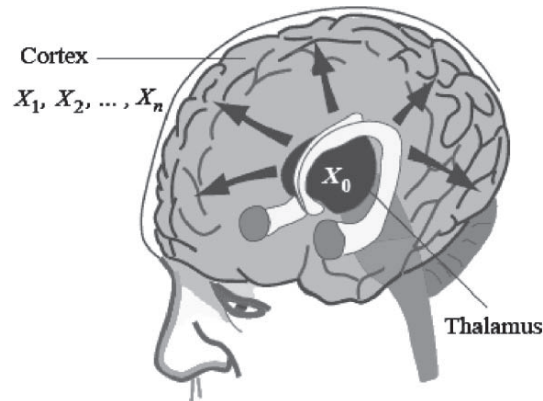


Fig. 1.2. A 1-to-many relation: $Thalamus \Rightarrow Cortex$ in the human brain (with permission from E. Izhikevich).

1.1.1 Basics of Brain Physiology

The nervous system consists basically of two types of cells: neurons and glia (see Appendix for a glossary of neurobiological terms). *Neurons* (also called nerve cells, see Figure 1.3) are the primary cells, morphologic and functional units of the nervous system. They are found in the brain, the spinal cord and in the peripheral nerves and ganglia. Neurons consist of four major parts,

including the *dendrites* (shorter projections), which are responsible for receiving stimuli; the *axon* (longer projection), which sends the nerve impulse away from the cell; the *cell body*, which is the site of metabolic activity in the cell; and the *axon terminals*, which connect neurons to other neurons, or neurons to other body structures. Each neuron can have several hundred axon terminals that attach to another neuron multiple times, attach to multiple neurons, or both. Some types of neurons, such as Purkinje cells, have over 1000 dendrites. The body of a neuron, from which the axon and dendrites project, is called the *soma* and holds the nucleus of the cell. The nucleus typically occupies most of the volume of the soma and is much larger in diameter than the axon and dendrites, which typically are only about a micrometer thick or less. Neurons join to one another and to other cells through synapses.

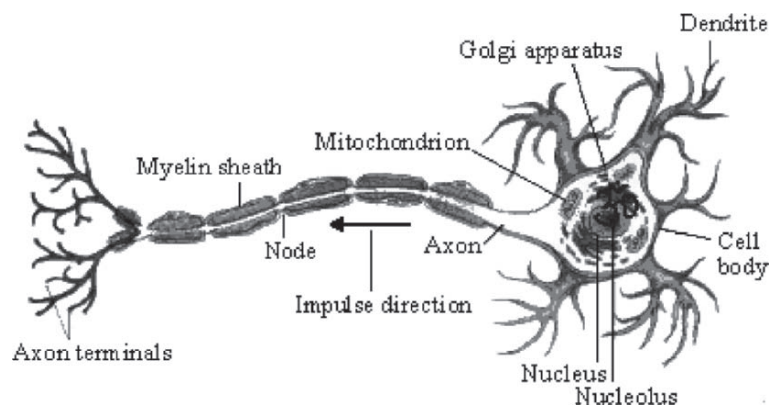


Fig. 1.3. A typical neuron, containing all of the usual cell organelles. However, it is highly specialized for the conductance of nerve impulse.

A defining feature of neurons is their ability to become ‘electrically excited’, that is, to undergo an action potential—and to convey this excitation rapidly along their axons as an impulse. The narrow cross section of axons and dendrites lessens the metabolic expense of conducting action potentials, although fatter axons convey the impulses more rapidly, generally speaking.

Many neurons have insulating sheaths of myelin around their axons, which enable their action potentials to travel faster than in unmyelinated axons of the same diameter. Formed by glial cells, the myelin sheathing normally runs along the axon in sections about 1 mm long, punctuated by unsheathed nodes of Ranvier. Neurons and glia make up the two chief cell types of the nervous system.

An action potential that arrives at its terminus in one neuron may provoke an action potential in another through release of neurotransmitter molecules across the synaptic gap.

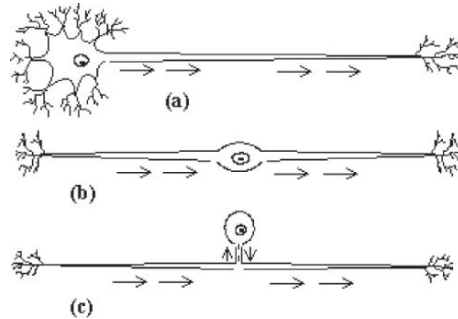


Fig. 1.4. Three structural classes of human neurons: (a) multipolar, (b) bipolar, and (c) unipolar.

There are three structural classes of neurons in the human body (see Figure 1.4):

1. The *multipolar neurons*, the majority of neurons in the body, in particular in the central nervous system.
2. The *bipolar neurons*, sensory neurons found in the special senses.
3. The *unipolar neurons*, sensory neurons located in dorsal root ganglia.

Neuronal Circuits

Figure 1.5 depicts a general model of a convergent circuit, showing two neurons converging on one neuron. This allows one neuron or *neuronal pool* to receive input from multiple sources. For example, the neuronal pool in the brain that regulates rhythm of breathing receives input from other areas of the brain, baroreceptors, chemoreceptors, and stretch receptors in the lungs.

Glia are specialized cells of the nervous system whose main function is to ‘glue’ neurons together. Specialized glia called *Schwann cells* secrete myelin sheaths around particularly long axons. Glia of the various types greatly outnumber the actual neurons.

The human nervous system consists of the central and peripheral parts (see Figure 1.6). The *central nervous system* (CNS) refers to the core nervous system, which consists of the *brain* and *spinal cord* (as well as *spinal nerves*). The *peripheral nervous system* (PNS) consists of the nerves and neurons that reside or extend outside the central nervous system—to serve the limbs and organs, for example. The peripheral nervous system is further divided into the somato-motoric nervous system and the autonomic nervous system (see Figure 1.7).

The CNS is further divided into two parts: the *brain* and the *spinal cord*. The average adult human brain weighs 1.3 to 1.4 kg (approximately 3 pounds). The brain contains about 100 billion nerve cells (neurons) and trillions of ‘support cells’ called glia. Further divisions of the human brain are depicted

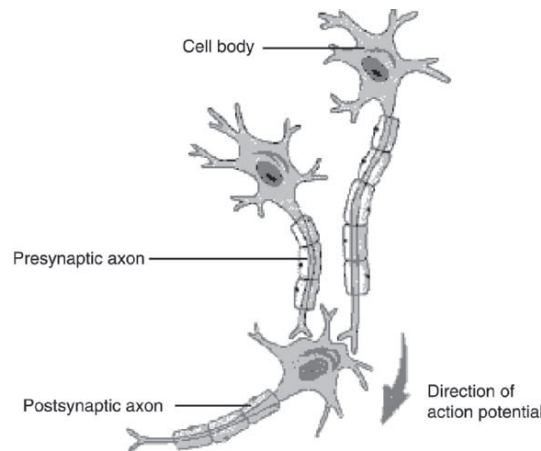


Fig. 1.5. A convergent neural circuit: nerve impulses arriving at the same neuron.

in Figure 1.8. The spinal cord is about 43 cm long in adult women and 45 cm long in adult men and weighs about 35-40 grams. The vertebral column, the collection of bones (back bone) that houses the spinal cord, is about 70 cm long. Therefore, the spinal cord is much shorter than the vertebral column.

The PNS is further divided into two major parts: the *somatic nervous system* and the *autonomic nervous system*.

The somatic nervous system consists of *peripheral nerve fibers* that send sensory information to the central nervous system and *motor nerve fibers* that project to skeletal muscle.

The autonomic nervous system (ANS) controls smooth muscles of the viscera (internal organs) and glands. In most situations, we are unaware of the workings of the ANS because it functions in an involuntary, reflexive manner. For example, we do not notice when blood vessels change size or when our heart beats faster. The ANS is most important in two situations:

1. In *emergencies* that cause stress and require us to ‘fight’ or take ‘flight’, and
2. In *non-emergencies* that allow us to ‘rest’ and ‘digest’.

The ANS is divided into three parts:

1. The *sympathetic nervous system* (see Figure 1.9),
2. The *parasympathetic nervous system* (see Figure 1.10), and
3. The *enteric nervous system*, which is a meshwork of nerve fibers that innervate the viscera (gastrointestinal tract, pancreas, gall bladder).

In the PNS, neurons can be functionally divided in 3 ways:

1. • *sensory (afferent)* neurons – carry information into the CNS from sense organs, and

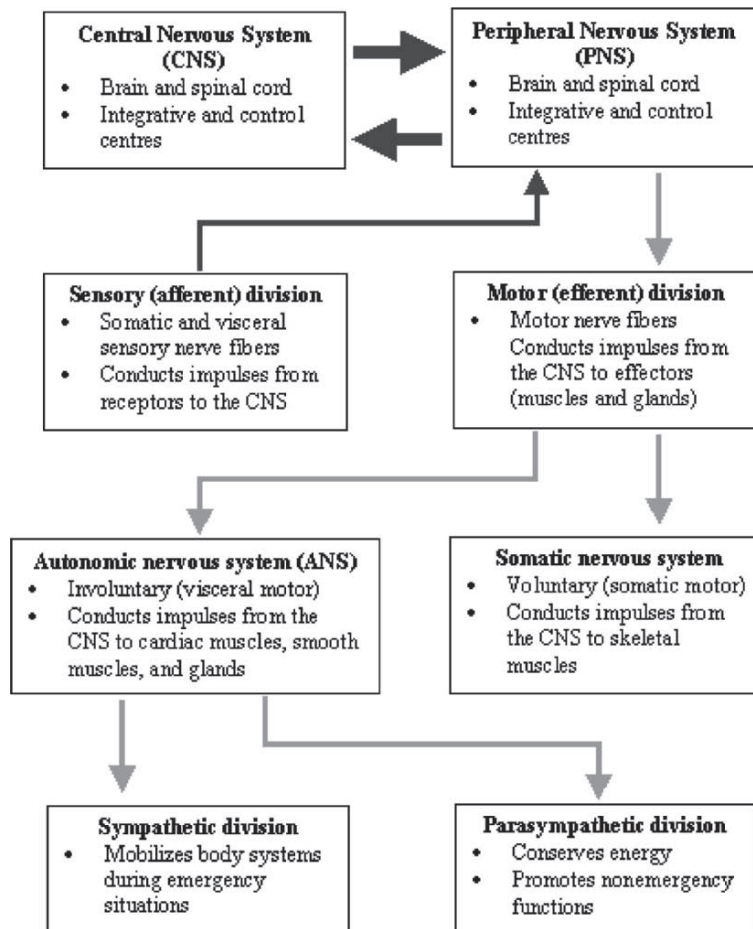


Fig. 1.6. Organization of the human nervous system.

- *motor (efferent)* neurons – carry information away from the CNS (for muscle control).
- 2. • *cranial* neurons – connect the brain with the periphery, and
- *spinal* neurons – connect the spinal cord with the periphery.
- 3. • *somatic* neurons – connect the skin or muscle with the central nervous system, and
- *visceral* neurons – connect the internal organs with the central nervous system.

Some differences between the PNS and the CNS are:

1. • In the CNS, collections of neurons are called *nuclei*.
- In the PNS, collections of neurons are called *ganglia*.
2. • In the CNS, collections of axons are called *tracts*.

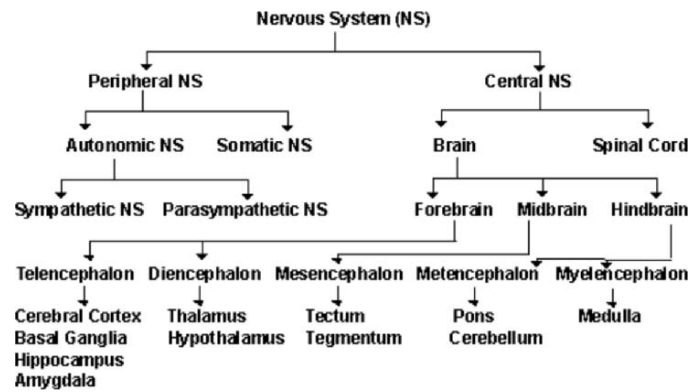


Fig. 1.7. Basic divisions of the human nervous system.

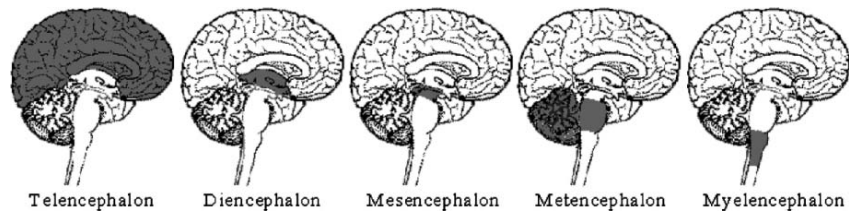


Fig. 1.8. Basic divisions of the human Brain.

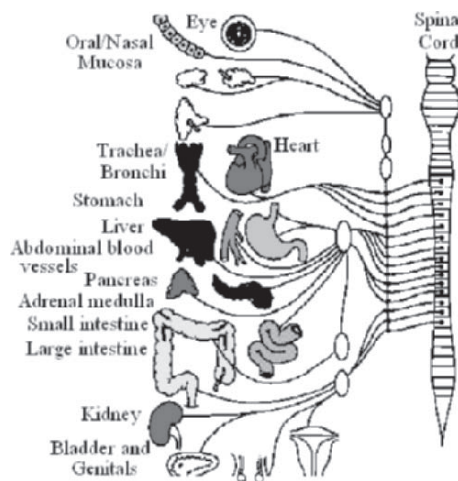


Fig. 1.9. Basic functions of the sympathetic nervous system.

- In the PNS, collections of axons are called *nerves*.

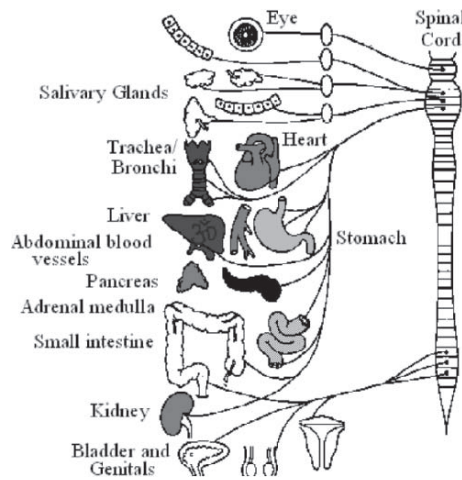


Fig. 1.10. Basic functions of the parasympathetic nervous system.

Basic Brain Partitions and Their Functions

Cerebral Cortex. The word ‘cortex’ comes from the Latin word for ‘bark’ (of a tree). This is because the cortex is a sheet of tissue that makes up the outer layer of the brain. The thickness of the cerebral cortex varies from 2 to 6 mm. The right and left sides of the cerebral cortex are connected by a thick band of nerve fibers called the ‘corpus callosum’. In higher mammals such as humans, the cerebral cortex looks like it has many bumps and grooves. A bump or bulge on the cortex is called a gyrus (the plural of the word gyrus is ‘gyri’) and a groove is called a sulcus (the plural of the word sulcus is ‘sulci’). Lower mammals like rats and mice have very few gyri and sulci. The main cortical functions are: *thought, voluntary movement, language, reasoning, and perception.*

Cerebellum. The word ‘cerebellum’ comes from the Latin word for ‘little brain’. The cerebellum is located behind the brain stem. In some ways, the cerebellum is similar to the cerebral cortex: the cerebellum is divided into hemispheres and has a cortex that surrounds these hemispheres. Its main functions are: *movement, balance, and posture.*

Brain Stem. The brain stem is a general term for the area of the brain between the thalamus and spinal cord. Structures within the brain stem include the medulla, pons, tectum, reticular formation and tegmentum. Some of these areas are responsible for the most basic functions of life such as breathing, heart rate and blood pressure. Its main functions are: *breathing, heart rate, and blood pressure.*

Hypothalamus. The *hypothalamus* is composed of several different areas and is located at the base of the brain. Although it is the size of only a pea (about 1/300 of the total brain weight), the hypothalamus is responsible for

some very important functions. One important function of the hypothalamus is the control of body temperature. The hypothalamus acts like a ‘thermostat’ by sensing changes in body temperature and then sending signals to adjust the temperature. For example, if we are too hot, the hypothalamus detects this and then sends a signal to expand the capillaries in your skin. This causes blood to be cooled faster. The hypothalamus also controls the pituitary. Its main functions are: *body temperature, emotions, hunger, thirst, sexual instinct, and circadian rhythms*. The *hypothalamus* is ‘the boss’ of the ANS.

Thalamus. The thalamus receives sensory information and relays this information to the cerebral cortex. The cerebral cortex also sends information to the thalamus which then transmits this information to other areas of the brain and spinal cord. Its main functions are: *sensory processing and movement*.

Limbic System. The limbic system (or the limbic areas) is a group of structures that includes the amygdala, the hippocampus, mammillary bodies and cingulate gyrus. These areas are important for controlling the emotional response to a given situation. The hippocampus is also important for memory. Its main function is *emotions*.

Hippocampus. The hippocampus is one part of the limbic system that is important for memory and learning. Its main functions are: *learning and memory*.

Basal Ganglia. The basal ganglia are a group of structures, including the globus pallidus, caudate nucleus, subthalamic nucleus, putamen and substantia nigra, that are important in coordinating movement. Its main function is *movement*.

Midbrain. The midbrain includes structures such as the superior and inferior colliculi and red nucleus. There are several other areas also in the midbrain. Its main functions are: *vision, audition, eye movement, and body movement*.

Nerves

A *nerve* is an enclosed, cable-like bundle of *nerve fibers* or *axons*, which includes the glia that ensheath the axons in myelin (see [Mar98, GM88, II06a]).

Nerves are part of the peripheral nervous system. *Afferent nerves* convey sensory signals to the brain and spinal cord, for example from skin or organs, while *efferent* nerves conduct stimulatory signals from the *motor neurons* of the brain and spinal cord to the muscles and glands.

These signals, sometimes called nerve impulses, are also known as action potentials: Rapidly traveling electrical waves, which begin typically in the cell body of a neuron and propagate rapidly down the axon to its tip or terminus’.

Nerves may contain fibers that all serve the same purpose; for example *motor nerves*, the axons of which all terminate on muscle fibers and stimulate contraction. Or they be *mixed nerves*.

An *axon*, or ‘nerve fibre’, is a long slender projection of a nerve cell or neuron, which conducts electrical impulses away from the neuron’s cell body or soma. Axons are in effect the primary transmission lines of the nervous system, and as bundles they help make up nerves. The axons of many neurons are sheathed in myelin.

On the other hand, a *dendrite* is a slender, typically branched projection of a nerve cell or neuron, which conducts the electrical stimulation received from other cells through synapses to the body or soma of the cell from which it projects.

Many dendrites convey this stimulation passively, meaning without action potentials and without activation of voltage-gated ion channels. In such dendrites the voltage change that results from stimulation at a synapse may extend both towards and away from the soma. In other dendrites, though an action potential may not arise, nevertheless voltage-gated channels help to propagate excitatory synaptic stimulation. This propagation is efficient only toward the soma due to an uneven distribution of channels along such dendrites.

The structure and branching of a neuron’s dendrites strongly influences how it integrates the input from many others, particularly those that input only weakly (more at synapse). This integration is in aspects ‘temporal’—involving the summation of stimuli that arrive in rapid succession—as well as ‘spatial’—entailing the aggregation of excitatory and inhibitory inputs from separate branches or ‘arbors’.

Spinal nerves take their origins from the spinal cord. They control the functions of the rest of the body. In humans, there are 31 pairs of spinal nerves: 8 *cervical*, 12 *thoracic*, 5 *lumbar*, 5 *sacral* and 1 *coccygeal*.

Neural Action Potential

As the traveling signals of nerves and as the localized changes that contract muscle cells, *action potentials* are an essential feature of animal life. They set the pace of thought and action, constrain the sizes of evolving anatomies and enable centralized control and coordination of organs and tissues (see [Mar98, GM88, IS00]).

Basic Features

When a biological cell or patch of membrane undergoes an action potential, the polarity of the transmembrane voltage swings rapidly from negative to positive and back. Within any one cell, consecutive action potentials typically are indistinguishable. Also between different cells the amplitudes of the voltage swings tend to be roughly the same. But the speed and simplicity of action potentials vary significantly between cells, in particular between different cell types.

Minimally, an action potential involves a *depolarization*, a *re-polarization* and finally a *hyperpolarization* (or ‘undershoot’). In specialized muscle cells

of the heart, such as the pacemaker cells, a ‘plateau phase’ of intermediate voltage may precede re-polarization.

Underlying Mechanism

The transmembrane voltage changes that take place during an action potential result from changes in the permeability of the membrane to specific ions, the internal and external concentrations of which are in imbalance. In the axon fibers of nerves, depolarization results from the inward rush of sodium ions, while re-polarization and hyperpolarization arise from an outward rush of potassium ions. Calcium ions make up most or all of the depolarizing currents at an axon’s pre-synaptic terminus, in muscle cells and in some dendrites.

The imbalance of ions that makes possible not only action potentials but the resting cell potential arises through the work of pumps, in particular the sodium-potassium exchanger.

Changes in membrane permeability and the onset and cessation of ionic currents reflect the opening and closing of ‘voltage-gated’ ion channels, which provide portals through the membrane for ions. Residing in and spanning the membrane, these enzymes sense and respond to changes in transmembrane potential.

Initiation

Action potentials are triggered by an initial depolarization to the point of *threshold*. This threshold potential varies but generally is about 15 millivolts above the resting potential of the cell. Typically action potential initiation occurs at a synapse, but may occur anywhere along the axon. In his discovery of ‘animal electricity’, L. Galvani elicited an action potential through contact of his scalpel with the motor nerve of a frog he was dissecting, causing one of its legs to kick as in life.

Wave Propagation

In the fine fibers of simple (or ‘unmyelinated’) axons, action potentials propagate as waves, which travel at speeds up to 120 meters per second.

The propagation speed of these ‘impulses’ is faster in fatter fibers than in thin ones, other things being equal. In their *Nobel Prize* winning work uncovering the wave nature and ionic mechanism of action potentials, *Alan L. Hodgkin* and *Andrew F. Huxley* performed their celebrated experiments on the ‘giant fibre’ of Atlantic squid [HH52]. Responsible for initiating flight, this axon is fat enough to be seen without a microscope (100 to 1000 times larger than is typical). This is assumed to reflect an adaptation for speed. Indeed, the velocity of nerve impulses in these fibers is among the fastest in nature.

Saltatory propagation

Many neurons have insulating sheaths of myelin surrounding their axons, which enable action potentials to travel faster than in unmyelinated axons of the same diameter. The myelin sheathing normally runs along the axon in sections about 1 mm long, punctuated by unsheathed ‘nodes of Ranvier’.

Because the salty cytoplasm of the axon is electrically conductive, and because the myelin inhibits charge leakage through the membrane, depolarization at one node is sufficient to elevate the voltage at a neighboring node to the threshold for action potential initiation. Thus in myelinated axons, action potentials do not propagate as waves, but recur at successive nodes and in effect hop along the axon. This mode of propagation is known as *saltatory conduction*. Saltatory conduction is faster than smooth conduction. Some typical action potential velocities are as follows:

Fiber	Diameter	AP Velocity
Unmyelinated	0.2–1.0 micron	0.2–2 m/sec
Myelinated	2–20 microns	12–120 m/sec

The disease called *multiple sclerosis* (MS) is due to a breakdown of myelin sheathing, and degrades muscle control by destroying axons’ ability to conduct action potentials.

Detailed Features

Depolarization and re-polarization together are complete in about two milliseconds, while undershoots can last hundreds of milliseconds, depending on the cell. In neurons, the exact length of the roughly two-millisecond delay in re-polarization can have a strong effect on the amount of neurotransmitter released at a synapse. The duration of the hyperpolarization determines a nerve’s ‘refractory period’ (how long until it may conduct another action potential) and hence the frequency at which it will fire under continuous stimulation. Both of these properties are subject to biological regulation, primarily (among the mechanisms discovered so far) acting on ion channels selective for potassium.

A cell capable of undergoing an action potential is said to be *excitable*.

Synapses

Synapses are specialized junctions through which cells of the nervous system signal to one another and to non-neuronal cells such as muscles or glands (see Figure 1.11). Synapses define the circuits in which the neurons of the central nervous system interconnect. They are thus crucial to the biological computations that underlie perception and thought. They also provide the means through which the nervous system connects to and controls the other systems of the body (see [II06a]).

Anatomy and Structure. At a classical synapse, a mushroom-shaped bud projects from each of two cells and the caps of these buds press flat

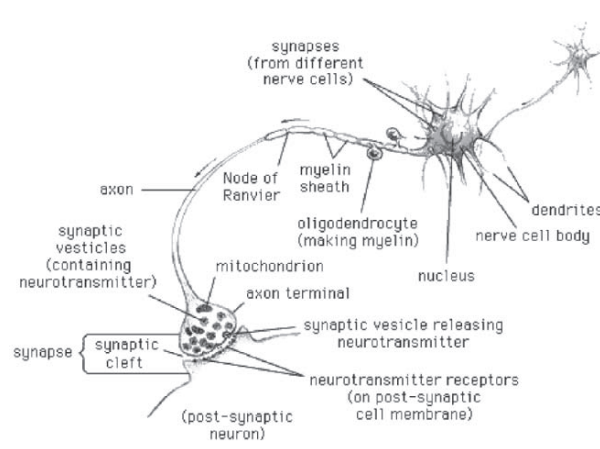


Fig. 1.11. Neuron forming a synapse.

against one another (see Figure 1.12). At this interface, the membranes of the two cells flank each other across a slender gap, the narrowness of which enables signaling molecules known as neurotransmitters to pass rapidly from one cell to the other by diffusion. This gap is sometimes called the synaptic cleft.

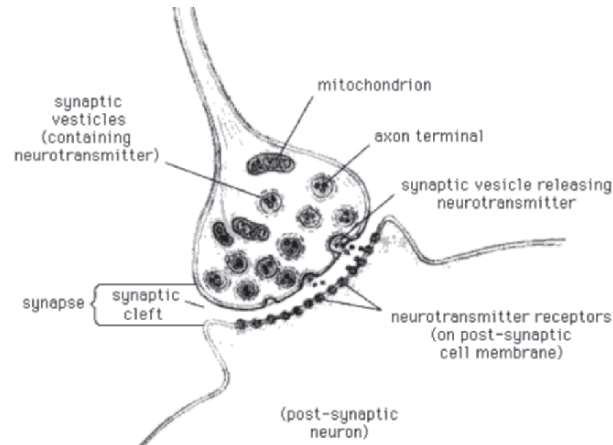


Fig. 1.12. Structure of a chemical synapse.

Synapses are asymmetric both in structure and in how they operate. Only the so-called pre-synaptic neuron secretes the neurotransmitter, which binds to receptors facing into the synapse from the post-synaptic cell. The pre-synaptic nerve terminal generally buds from the tip of an axon, while the

post-synaptic target surface typically appears on a dendrite, a cell body or another part of a cell.

Signalling across the Synapse

The release of neurotransmitter is triggered by the arrival of a nerve impulse (or action potential) and occurs through an unusually rapid process of cellular secretion: Within the pre-synaptic nerve terminal, vesicles containing neurotransmitter sit ‘docked’ and ready at the synaptic membrane. The arriving action potential produces an influx of calcium ions through voltage-dependent, calcium-selective ion channels, at which point the vesicles fuse with the membrane and release their contents to the outside. Receptors on the opposite side of the synaptic gap bind neurotransmitter molecules and respond by opening nearby ion channels in the post-synaptic cell membrane, causing ions to rush in or out and changing the local transmembrane potential of the cell. The result is *excitatory*, in the case of depolarizing currents, or *inhibitory* in the case of hyperpolarizing currents. Whether a synapse is excitatory or inhibitory depends on what type(s) of ion channel conduct the post-synaptic current, which in turn is a function of the type of receptors and neurotransmitter employed at the synapse.

Excitatory synapses in the brain show several forms of synaptic plasticity, including long-term potentiation (LTP) and long-term depression (LTD), which are initiated by increases in intracellular Ca^{2+} that are generated through NMDA (N-methyl-D-aspartate) receptors or voltage-sensitive Ca^{2+} channels. LTP depends on the coordinated regulation of an ensemble of enzymes, including Ca^{2+} /calmodulin-dependent protein kinase II, adenylyl cyclase 1 and 8, and calcineurin, all of which are stimulated by calmodulin, a Ca^{2+} -binding protein. [XS05] discussed the hypothesis that calmodulin is a central integrator of synaptic plasticity and that its unique regulatory properties allow the integration of several forms of signal transduction that are required for LTP and LTD.

Synaptic strength

The amount of current, or more strictly the change in transmembrane potential, depends on the ‘strength’ of the synapse, which is subject to biological regulation. One regulatory mechanism involves the simple coincidence of action potentials in the synaptically linked cells. Because the coincidence of sensory stimuli (the sound of a bell and the smell of meat, for example, in the experiments by *Nobel Laureate Ivan P. Pavlov*) can give rise to associative learning or conditioning, neuroscientists have hypothesized that synaptic strengthening through coincident activity in two neurons might underlie learning and memory. This is known as the *Hebbian theory* [Heb49]. It is related to *Pavlov’s conditional-reflex learning*: it is learning that takes place when we come to associate two stimuli in the environment. One of these stimuli triggers a reflexive response. The second stimulus is originally neutral with

respect to that response, but after it has been paired with the first stimulus, it comes to trigger the response in its own right.

Biophysics of Synaptic Transmission

Technically, synaptic transmission happens in transmitter-activated ion channels. Activation of a presynaptic neuron results in a release of neurotransmitters into the synaptic cleft. The transmitter molecules diffuse to the other side of the cleft and activate receptors that are located in the postsynaptic membrane. So-called ionotropic receptors have a direct influence on the state of an associated ion channel whereas metabotropic receptors control the state of the ion channel by means of a biochemical cascade of g -proteins and second messengers. In any case the activation of the receptor results in the opening of certain ion channels and, thus, in an excitatory or inhibitory postsynaptic current (EPSC or IPSC, respectively). The transmitter-activated ion channels can be described as an explicitly time-dependent conductivity $g_{syn}(t)$ that will open whenever a presynaptic spike arrives. The current that passes through these channels depends, as usual, on the difference of its reversal potential E_{syn} and the actual value of the membrane potential,

$$I_{syn}(t) = g_{syn}(t)(u - E_{syn}).$$

The parameter E_{syn} and the function $g_{syn}(t)$ can be used to characterize different types of synapse. Typically, a superposition of exponentials is used for $g_{syn}(t)$. For inhibitory synapses E_{syn} equals the reversal potential of potassium ions (about -75 mV), whereas for excitatory synapses $E_{syn} \approx 0$.

The effect of fast *inhibitory* neurons in the central nervous system of higher vertebrates is almost exclusively conveyed by a neuro-transmitter called γ -aminobutyric acid, or GABA for short. In addition to many different types of inhibitory interneurons, cerebellar Purkinje cells form a prominent example of projecting neurons that use GABA as their neuro-transmitter. These neurons synapse onto neurons in the deep cerebellar nuclei (DCN) and are particularly important for an understanding of cerebellar function.

The parameters that describe the conductivity of transmitter-activated ion channels at a certain synapse are chosen so as to mimic the time course and the amplitude of experimentally observed spontaneous postsynaptic currents. For example, the conductance $\bar{g}_{syn}(t)$ of inhibitory synapses in DCN neurons can be described by a simple exponential decay with a time constant of $\tau = 5$ ms and an amplitude of $\bar{g}_{syn} = 40$ pS,

$$g_{syn}(t) = \sum_f \bar{g}_{syn} \exp\left(-\frac{t - t^{(f)}}{\tau}\right) \Theta(t - t^{(f)}),$$

where $t^{(f)}$ denotes the arrival time of a presynaptic action potential. The reversal potential is given by that of potassium ions, viz. $E_{syn} = -75$ mV (see [GMK94]).

Clearly, more attention can be paid to account for the details of synaptic transmission. In cerebellar granule cells, for example, inhibitory synapses are also GABAergic, but their postsynaptic current is made up of two different components. There is a fast component, that decays with a time constant of about 5 ms, and there is a component that is ten times slower. The underlying postsynaptic conductance is thus of the form

$$g_{syn}(t) = \sum_f \left(\bar{g}_{fast} \exp\left(-\frac{t-t^{(f)}}{\tau_{fast}}\right) + \bar{g}_{slow} \exp\left(-\frac{t-t^{(f)}}{\tau_{slow}}\right) \right) \Theta(t-t^{(f)}).$$

Now, most of *excitatory* synapses in the vertebrate central nervous system rely on glutamate as their neurotransmitter. The postsynaptic receptors, however, can have very different pharmacological properties and often different types of glutamate receptors are present in a single synapse. These receptors can be classified by certain amino acids that may be selective agonists. Usually, NMDA (N-methyl-D-aspartate) and non-NMDA receptors are distinguished. The most prominent among the non-NMDA receptors are AMPA-receptors. Ion channels controlled by AMPA-receptors are characterized by a fast response to presynaptic spikes and a quickly decaying postsynaptic current. NMDA-receptor controlled channels are significantly slower and have additional interesting properties that are due to a voltage-dependent blocking by magnesium ions (see [GMK94]).

Excitatory synapses in cerebellar granule cells, for example, contain two different types of glutamate receptors, viz. AMPA- and NMDA-receptors. The time course of the postsynaptic conductivity caused by an activation of AMPA-receptors at time $t = t^{(f)}$ can be described as follows,

$$g_{AMPA}(t) = \bar{g}_{AMPA} \cdot N \cdot \left(\exp\left(-\frac{t-t^{(f)}}{\tau_{decay}}\right) - \exp\left(-\frac{t-t^{(f)}}{\tau_{rise}}\right) \right) \Theta(t-t^{(f)}),$$

with rise time $\tau_{rise} = 0.09$ ms, decay time $\tau_{decay} = 1.5$ ms, and maximum conductance $\bar{g}_{AMPA} = 720$ pS. The numerical constant $N = 1.273$ normalizes the maximum of the braced term to unity (see [GMK94]).

NMDA-receptor controlled channels exhibit a significantly richer repertoire of dynamic behavior because their state is not only controlled by the presence or absence of their agonist, but also by the membrane potential. The voltage dependence itself arises from the blocking of the channel by a common extracellular ion, Mg^{2+} . Unless Mg^{2+} is removed from the extracellular medium, the channels remain closed at the resting potential even in the presence of NMDA. If the membrane is depolarized beyond -50 mV, then the Mg^{2+} -block is removed, the channel opens, and, in contrast to AMPA-controlled channels, stays open for 10–100 milliseconds. A simple ansatz that accounts for this additional voltage dependence of NMDA-controlled channels in cerebellar granule cells is

$$g_{NMDA}(t) = \bar{g}_{NMDA} \cdot N \cdot \left(\exp\left(-\frac{t - t^{(f)}}{\tau_{decay}}\right) - \exp\left(-\frac{t - t^{(f)}}{\tau_{rise}}\right) \right) g_{\infty} \Theta(t - t^{(f)}),$$

where $g_{\infty} = (1 + e^{\alpha u} [\text{Mg}^{2+}]_o / \beta)$, $\tau_{rise} = 3$ ms, $\tau_{decay} = 40$ ms, $N = 1.358$, $\bar{g}_{NMDA} = 1.2$ nS, $\alpha = 0.062$ mV⁻¹, $\beta = 3.57$ mM, and the extracellular magnesium concentration $[\text{Mg}^{2+}]_o = 1.2$ mM (see [GMK94]).

Finally, Though NMDA-controlled ion channels are permeable to sodium and potassium ions, their permeability to Ca^{2+} is even five or ten times larger. Calcium ions are known to play an important role in intracellular signaling and are probably also involved in long-term modifications of synaptic efficacy. Calcium influx through NMDA-controlled ion channels, however, is bound to the coincidence of presynaptic (NMDA release from presynaptic sites) and postsynaptic (removal of the Mg^{2+} -block) activity. Hence, NMDA-receptors operate as a kind of a molecular coincidence detectors as they are required for a biochemical implementation of Hebb's learning rule [Heb49].

Reflex Action: the Basis of CNS Activity

The basis of all CNS activity, as well as the simplest example of our sensory-motor adjunction, is the *reflex (sensory-motor) action*, RA . It occurs at all neural organizational levels. We are aware of some reflex acts, while others occur without our knowledge.

In particular, the spinal reflex action is defined as a composition of neural pathways, $RA = EN \circ CN \circ AN$, where EN is the efferent neuron, AN is the afferent neuron and $CN = CN_1, \dots, CN_n$ is the chain of n connector neurons ($n = 0$ for the simplest, stretch, reflex, $n \geq 1$ for all other reflexes). In other words, the following diagram commutes:

$$\begin{array}{ccc} PSC & \xrightarrow{CN} & ASC \\ AN \uparrow & & \downarrow EN \\ Rec & \xrightarrow{RA} & Eff \end{array}$$

in which Rec is the receptor (e.g., eye), Eff is the effector (e.g., muscle), PSC is the posterior (or, dorsal) horn of the spinal cord, and ASC is the anterior (or, ventral) horn of the spinal cord. In this way defined map $RA : Rec \rightarrow Eff$ is the simplest, *one-to-one* relation between one receptor neuron and one effector neuron (e.g., patellar reflex, see Figure 1.13).

Now, in the majority of human reflex arcs a chain CN of many connector neurons is found. There may be link-ups with various levels of the brain and spinal cord. Every receptor neuron is potentially linked in the CNS with a large number of effector organs all over the body, i.e., the map $RA : Rec \rightarrow Eff$ is *one-to-many*. Similarly, every effector neuron is potentially in communication

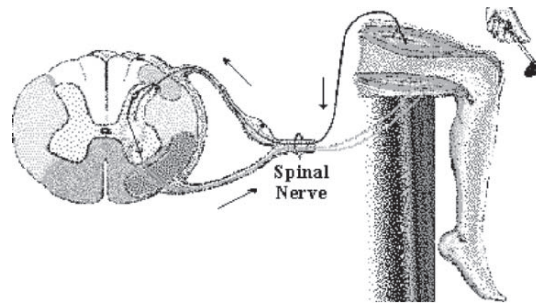


Fig. 1.13. Schematic of a simple knee-jerk reflex. Hammer strikes knee, generating sensory impulse to spinal cord. Primary neuron makes (monosynaptic, excitatory) synapse with anterior horn (motor) cell, whose axon travels via ventral root to quadriceps muscle, which contracts, raising foot. Hamstring (lower) muscle is simultaneously inhibited, via an interneuron.

with receptors all over the body, i.e., the map $RA : Rec \rightarrow Eff$ is *many-to-one*.

However, the most frequent form of the map $RA : Rec \rightarrow Eff$ is *many-to-many*. Other neurons synapsing with the effector neurons may give a complex link-up with centers at higher and lower levels of the CNS. In this way, higher centers in the brain can modify reflex acts which occur through the spinal cord. These centers can send suppressing or facilitating impulses along their pathways to the cells in the spinal cord [II06a].

Through such ‘functional’ link-ups, neurons in different parts of the CNS, when active, can influence each other. This makes it possible for Pavlov’s conditioned reflexes to be established. Such reflexes form the basis of all training, so that it becomes difficult to say where reflex (or involuntary) behavior ends and purely voluntary behavior begins [Ano80].

In particular, the control of voluntary movements is extremely complex. Many different systems across numerous brain areas need to work together to ensure proper motor control. Our understanding of the nervous system decreases as we move up to higher CNS structures.

A Bird’s Look at the Brain

The *brain* is the supervisory center of the nervous system consisting of *grey matter* (superficial parts called *cortex* and deep brain nuclei) and *white matter* (deep parts except the brain nuclei). It controls and coordinates behavior, *homeostasis*¹ (i.e., negative feedback control of the body functions such as heartbeat, blood pressure, fluid balance, and body temperature) and mental

¹ *Homeostasis* is the property of an open system to regulation its internal environment so as to maintain a stable state of structure and functions, by means of multiple dynamic equilibrium controlled by interrelated regulation mechanisms. The term was coined in 1932 by W. Cannon from two Greek words [*homeo*–*man*]

functions (such as cognition, emotion, memory and learning) (see [Mar98, GM88, II06a]).

The *vertebrate brain* can be subdivided into: (i) *medulla oblongata* (or, *brain stem*); (ii) *myelencephalon*, divided into: *pons* and *cerebellum*; (iii) *mesencephalon* (or, *midbrain*); (iv) *diencephalon*; and (v) *telencephalon* (*cerebrum*).

Sometimes a gross division into three major parts is used: *hindbrain* (including medulla oblongata and myelencephalon), *midbrain* (mesencephalon) and *forebrain* (including diencephalon and telencephalon). The cerebrum and the cerebellum consist each of two *hemispheres*. The *corpus callosum* connects the two hemispheres of the cerebrum.

The cerebrum and the cerebellum consist each of two hemispheres. The corpus callosum connects the two hemispheres of the cerebrum. The cerebellum is a cauliflower-shaped section of the brain (see Figure 1.14). It is located in the hindbrain, at the bottom rear of the head, directly behind the pons. The cerebellum is a complex computer mostly dedicated to the intricacies of managing walking and balance. Damage to the cerebellum leaves the sufferer with a gait that appears drunken and is difficult to control.

The spinal cord is the extension of the central nervous system that is enclosed in and protected by the vertebral column. It consists of nerve cells and their connections (axons and dendrites), with both gray matter and white matter, with the former surrounded by the latter.

Cranial nerves are nerves which start directly from the brainstem instead of the spinal cord, and mainly control the functions of the anatomic structures of the head. In human anatomy, there are exactly *12 pairs* of them: (I) *olfactory nerve*, (II) *optic nerve*, (III) oculomotor nerve, (IV) Trochlear nerve, (V) Trigeminal nerve, (VI) Abducens nerve, (VII) Facial nerve, (VIII) Vestibulo-

and [*stasis-stationary*]. Homeostasis is one of the fundamental characteristics of living things. It is the maintenance of the internal environment within tolerable limits. All sorts of factors affect the suitability of our body fluids to sustain life; these include properties like temperature, salinity, acidity (carbon dioxide), and the concentrations of nutrients and wastes (urea, glucose, various ion, oxygen). Since these properties affect the chemical reactions that keep bodies alive, there are built-in physiological mechanisms to maintain them at desirable levels. This control is achieved with various organs and glands in the body. For example [Mar98, GM88, II06a]: The hypothalamus monitors water content, carbon dioxide concentration, and blood temperature, sending nerve impulses to the pituitary gland and skin. The pituitary gland synthesizes ADH (anti-diuretic hormone) to control water content in the body. The muscles can shiver to produce heat if the body temperature is too low. Warm-blooded animals (homeotherms) have additional mechanisms of maintaining their internal temperature through homeostasis. The pancreas produces insulin to control blood-sugar concentration. The lungs take in oxygen and give out carbon dioxide. The kidneys remove urea and adjust ion and water concentrations. More realistic is dynamical homeostasis, or *homeokinesis*, which forms the basis of the *Anochin's theory of functional systems* [Ano80]

cochlear nerve (sometimes called the auditory nerve), (IX) Glossopharyngeal nerve, (X) Vagus nerve, (XI) Accessory nerve (sometimes called the spinal accessory nerve), and (XII) Hypoglossal nerve.

The optic nerve consists mainly of axons extending from the ganglionic cells of the eye's retina. The axons terminate in the lateral geniculate nucleus, pulvinar, and superior colliculus, all of which belong to the primary visual center. From the lateral geniculate body and the pulvinar fibers pass to the visual cortex.

In particular, the *optic nerve* contains roughly one million nerve fibers. This number is low compared to the roughly 130 million receptors in the retina, and implies that substantial pre-processing takes place in the retina before the signals are sent to the brain through the optic nerve.

In most vertebrates the mesencephalon is the highest integration center in the brain, whereas in mammals this role has been adopted by the telencephalon. Therefore the cerebrum is the largest section of the mammalian brain and its surface has many deep fissures (sulci) and grooves (gyri), giving an excessively wrinkled appearance to the brain.

The *human brain* can be subdivided into several distinct regions:

The *cerebral hemispheres* form the largest part of the brain, occupying the anterior and middle cranial fossae in the skull and extending backwards over the tentorium cerebelli. They are made up of the cerebral cortex, the basal ganglia, tracts of synaptic connections, and the ventricles containing CSF.

The *diencephalon* includes the thalamus, hypothalamus, epithalamus and subthalamus, and forms the central core of the brain. It is surrounded by the cerebral hemispheres.

The *midbrain* is located at the junction of the middle and posterior cranial fossae.

The *pons* sits in the anterior part of the posterior cranial fossa; the fibres within the structure connect one cerebral hemisphere with its opposite cerebellar hemisphere.

The *medulla oblongata* is continuous with the spinal cord, and is responsible for automatic control of the respiratory and cardiovascular systems.

The *cerebellum* overlies the pons and medulla, extending beneath the tentorium cerebelli and occupying most of the posterior cranial fossa. It is mainly concerned with motor functions that regulate muscle tone, coordination, and posture.

Now, the two *cerebral hemispheres* (see Figure 1.14) can be further divided into *four lobes*:

The *frontal lobe* is concerned with higher intellectual functions, such as abstract thought and reason, speech (Broca's area in the left hemisphere only), olfaction, and emotion. Voluntary movement is controlled in the precentral gyrus (the primary motor area, see Figure 1.15).

The *parietal lobe* is dedicated to sensory awareness, particularly in the postcentral gyrus (the primary sensory area, see Figure 1.15). It is also as-

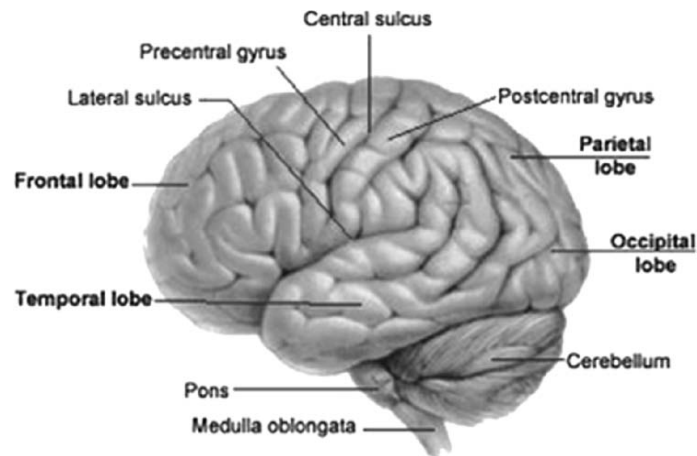


Fig. 1.14. The human cerebral hemispheres.

sociated with abstract reasoning, language interpretation and formation of a mental egocentric map of the surrounding area.

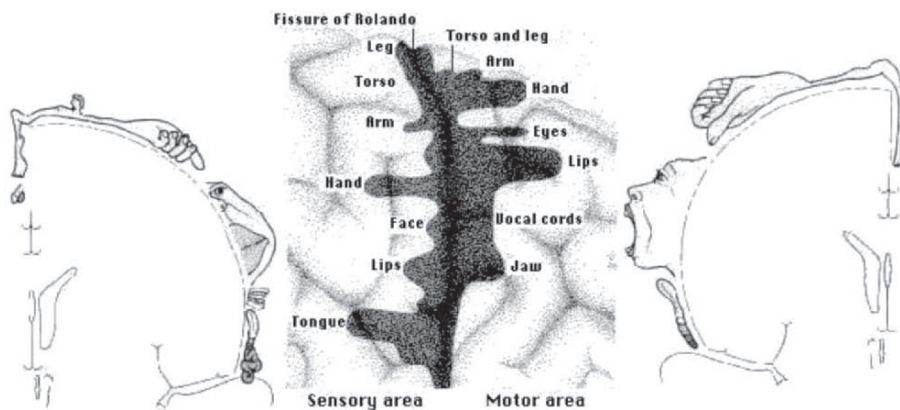


Fig. 1.15. Penfield's 'Homunculus', showing the primary somatosensory and motor areas of the human brain.

The *occipital lobe* is responsible for interpretation and processing of visual stimuli from the optic nerves, and association of these stimuli with other nervous inputs and memories.

The *temporal lobe* is concerned with emotional development and formation, and also contains the auditory area responsible for processing and discrimination of sound. It is also the area thought to be responsible for the formation and processing of memories.

1.1.2 Modern 3D Brain Imaging

Nuclear Magnetic Resonance and 2D Brain Imaging

The *Nobel Prize in Physiology or Medicine in 2003* was jointly awarded to *Paul C. Lauterbur* and *Peter Mansfield* for their discoveries concerning *magnetic resonance imaging* (MRI), a technique for using strong magnetic fields to produce images of the inside of the human body.

Atomic nuclei in a strong magnetic field rotate with a frequency that is dependent on the strength of the magnetic field. Their energy can be increased if they absorb radio waves with the same resonant frequency. When the atomic nuclei return to their previous energy level, radio waves are emitted. These discoveries were awarded the *Nobel Prize in Physics in 1952*, jointly to *Felix Bloch* and *Edward M. Purcell*. During the following decades, magnetic resonance was used mainly for studies of the chemical structure of substances. In the beginning of the 1970s, Lauterbur and Mansfield made pioneering contributions, which later led to the applications of nuclear magnetic resonance (NMR) in medical imaging.

Paul Lauterbur discovered the possibility to create a 2D picture by introducing gradients in the magnetic field. By analysis of the characteristics of the emitted radio waves, he could determine their origin. This made it possible to build up 2D pictures of structures that could not be visualized with other methods.

Peter Mansfield further developed the utilization of gradients in the magnetic field. He showed how the signals could be mathematically analyzed, which made it possible to develop a useful imaging technique. Mansfield also showed how extremely fast imaging could be achievable. This became technically possible within medicine a decade later.

Magnetic resonance imaging (MRI), is now a routine method within medical diagnostics. Worldwide, more than 60 million investigations with MRI are performed each year, and the method is still in rapid development. MRI is often superior to other imaging techniques and has significantly improved diagnostics in many diseases. MRI has replaced several invasive modes of examination and thereby reduced the risk and discomfort for many patients.

3D MRI of Human Brain

Modern technology of human brain imaging emphasizes 3D investigation of brain structure and function, using three variations of MRI. Brain *structure* is commonly imaged using *anatomical MRI*, or aMRI, while brain *physiology* is usually imaged using *functional MRI*, or fMRI. For bridging the gap between brain anatomy and function, as well as exploring natural brain connectivity, a *diffusion MRI*, or dMRI is used, based on the state-of-the-art *diffusion tensor* (DT) technique (see [BJH95, Bih96, Bih00, BMP01, BZ02, Bih03]).

The ability to visualize anatomical connections between different parts of the brain, non-invasively and on an individual basis, has opened a new

era in the field of functional neuro-imaging. This major breakthrough for neuroscience and related clinical fields has developed over the past ten years through the advance of *diffusion magnetic resonance imaging* (dMRI). The concept of dMRI is to produce MRI quantitative maps of microscopic, natural displacements of water molecules that occur in brain tissues as part of the physical diffusion process. Water molecules are thus used as a probe that can reveal microscopic details about tissue architecture, either normal or in a diseased state.

Molecular Diffusion in a 3D Brain Volume

Molecular diffusion refers to the *Brownian motion* of molecules (see section 4.6 above), which results from the thermal energy carried by these molecules. Molecules travel randomly in space over a distance that is statistically well described by a *diffusion coefficient*, D . This coefficient depends only on the size (mass) of the molecules, the temperature and the nature (viscosity) of the medium (see Figure 1.16).

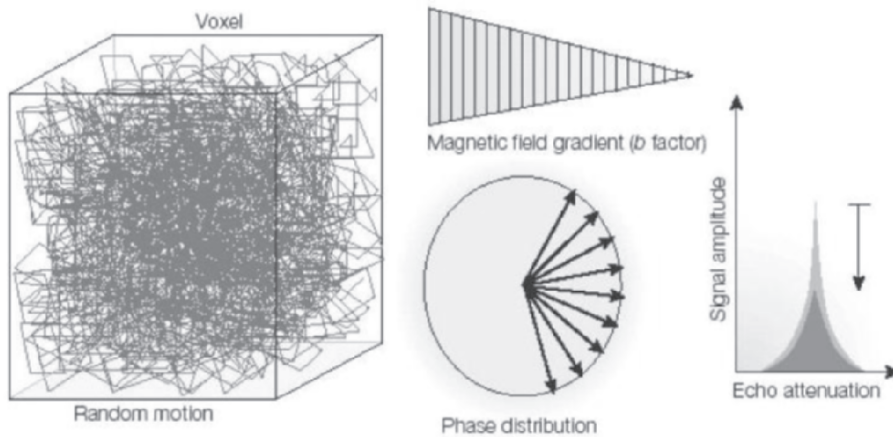


Fig. 1.16. Principles of dMRI: In the spatially varying magnetic field, induced through a magnetic field gradient, the amplitude and timing of which is characterized by a b -factor, moving molecules emit radiofrequency signals with slightly different phases. In a small 3D-volume (voxel) containing a large number of diffusing molecules, these phases become randomly distributed, directly reflecting the diffusion process, i.e., the trajectory of individual molecules. This diffusive phase distribution of the signal results in an *attenuation* A of the MRI signal, which quantitatively depends on the gradient characteristics of the b -factor and the diffusion coefficient D (adapted from [Bih03]).

dMRI is, thus, deeply rooted in the concept that, during their diffusion-driven displacements, molecules probe tissue structure at a *microscopic scale*

well beyond the usual *millimetric image resolution*. During typical diffusion times of about 50–100 ms, water molecules move in brain tissues on average over distances around 1–15 μm , bouncing, crossing or interacting with many tissue components, such as cell membranes, fibres or macromolecules. Because of the tortuous movement of water molecules around those obstacles, the actual diffusion distance is reduced compared to free water. Hence, the non-invasive observation of the water diffusion-driven displacement distributions *in vivo* provides unique clues to the fine structural features and geometric organization of neural tissues, and to changes in those features with physiological or pathological states.

Imaging Brain Diffusion with MRI

While early water diffusion measurements were made in biological tissues using Nuclear Magnetic Resonance in the 1960s and 70s, it is not until the mid 1980s that the basic principles of dMRI were laid out. MRI signals can be made sensitive to diffusion through the use of a pair of sharp magnetic field gradient pulses, the duration and the separation of which can be adjusted. The result is a signal (echo) attenuation which is precisely and quantitatively linked to the amplitude of the molecular displacement distribution: Fast (slow) diffusion results in a large (small) distribution and a large (small) signal attenuation. Naturally, the effect also depends on the intensity of the magnetic field gradient pulses.

In practice, any MRI imaging technique can be sensitized to diffusion by inserting the adequate magnetic field gradient pulses. By acquiring data with various gradient pulse amplitudes one gets images with different degrees of diffusion sensitivity (see Figure 1.17). Contrast in these images depends on diffusion, but also on other MRI parameters, such as the water relaxation times. Hence, these images are often numerically combined to determine, using a global diffusion model, an estimate of the diffusion coefficient in each image location. The resulting images are maps of the diffusion process and can be visualized using a quantitative scale.

Because the overall signal observed in a MRI image voxel, at a millimetric resolution, results from the integration, on a statistical basis, of all the microscopic displacement distributions of the water molecules present in this voxel it was suggested [6] to portray the complex diffusion processes that occur in a biological tissue on a voxel scale using a global, statistical parameter, the *apparent diffusion coefficient* (ADC). The ADC concept has been largely used since then in the literature. The ADC now depends not only on the actual diffusion coefficients of the water molecular populations present in the voxel, but also on experimental, technical parameters, such as the voxel size and the diffusion time.

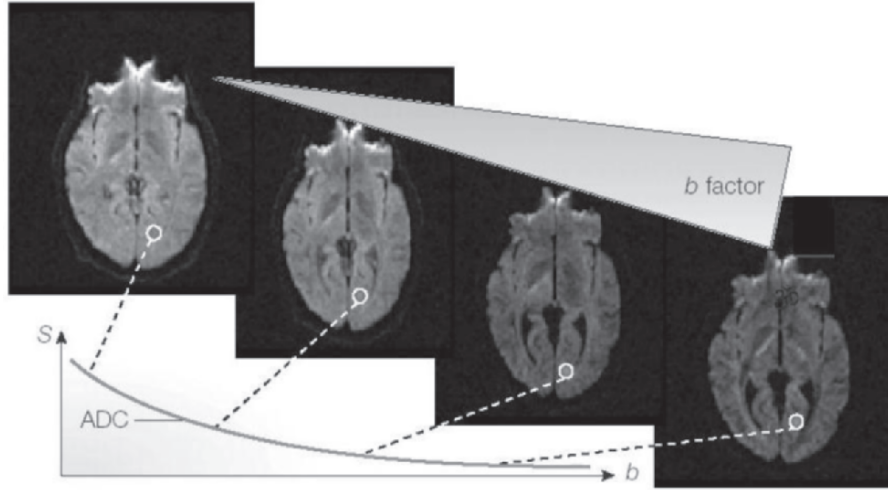


Fig. 1.17. Different degrees of *diffusion-weighted images* can be obtained using different values of the b -factor. The larger the b -factor the more the signal intensity becomes attenuated in the image. This attenuation, though, is modulated by the diffusion coefficient: signal in structures with fast diffusion (e.g., water filled ventricular cavities) decays very fast with b , while signal in tissues with low diffusion (e.g., gray and white matter) decreases more slowly. By fitting the signal decay as a function of b , one obtains the apparent diffusion coefficient (ADC) for each elementary volume (voxel) of the image. Calculated diffusion images (ADC maps), depending solely on the diffusion coefficient, can then be generated and displayed using a gray (or color) scale: High diffusion, as in the ventricular cavities, appears bright, while low diffusion is dark (adapted from [Bih03]).

3D Diffusion Brain Tensor

Now, as diffusion is really a 3D process, water molecular mobility in tissues is not necessarily the same in all directions. This *diffusion anisotropy* may result from the presence of obstacles that limit molecular movement in some directions. It is not until the advent of diffusion MRI that anisotropy was detected for the first time *in vivo*, at the end of the 1980s, in spinal cord and brain white matter (see [BJH95, Bih96, Bih00, BMP01, BZ02, Bih03]). Diffusion anisotropy in white matter grossly originates from its specific organization in bundles of more or less myelinated axonal fibres running in parallel: Diffusion in the direction of the fibres (whatever the species or the fiber type) is about 3–6 times faster than in the perpendicular direction. However the relative contributions of the intra-axonal and extracellular spaces, as well as the presence of the myelin sheath, to the ADC, and the exact mechanism for the anisotropy is still not completely understood, and remains the object of active research. It quickly became apparent, however, that this anisotropy effect could be exploited to map out the orientation in space of the white matter tracks in the

brain, assuming that the direction of the fastest diffusion would indicate the overall orientation of the fibres. The work on diffusion anisotropy really took off with the introduction in the field of diffusion MRI of the more rigorous formalism of the diffusion tensor.

More precisely, with plain diffusion MRI, diffusion is fully described using a single (scalar) parameter, the diffusion coefficient, D . The effect of diffusion on the MRI signal (most often a spin-echo signal) is an attenuation, A , which depends on D and on the b factor, which characterizes the gradient pulses (timing, amplitude, shape) used in the MRI sequence:

$$A = \exp(-bD).$$

However, in the presence of anisotropy, diffusion can no longer be characterized by a single scalar coefficient, but requires a 3D tensor-field $\mathbf{D} = \mathbf{D}(t)$ (see Chapter 3 above), give by the matrix of ‘moments’ (on the main diagonal) and ‘product’ (off-diagonal elements) [BMP01]:

$$\mathbf{D}(t) = \begin{pmatrix} D_{xx}(t) & D_{xy}(t) & D_{xz}(t) \\ D_{yx}(t) & D_{yy}(t) & D_{yz}(t) \\ D_{zx}(t) & D_{zy}(t) & D_{zz}(t) \end{pmatrix},$$

which fully describes molecular mobility along each direction and correlation between these directions. This tensor is symmetric ($D_{ij} = D_{ji}$, with $i, j = x, y, z$).

Now, in a *reference frame* $[x', y', z']$ that coincides with the principal or self directions of diffusivity, the off-diagonal terms do not exist, and the tensor is reduced only to its diagonal terms, $\{D_{x'x'}, D_{y'y'}, D_{z'z'}\}$, which represent molecular mobility along axes x' , y' , and z' , respectively. The *echo attenuation* then becomes:

$$A = \exp(-b_{x'x'}D_{x'x'} - b_{y'y'}D_{y'y'} - b_{z'z'}D_{z'z'}),$$

where b_{ii} are the elements of the **b**-*tensor*, which now replaces the scalar b -factor, expressed in the coordinates of this reference frame.

In practice, unfortunately, measurements are made in the reference frame $[x, y, z]$ of the MRI scanner gradients, which usually does not coincide with the diffusion frame of the tissue [BMP01]. Therefore, one must also consider the coupling of the nondiagonal elements, b_{ij} , of the **b**-tensor with the nondiagonal terms, D_{ji} , ($i \neq j$), of the diffusion tensor (now expressed in the scanner frame), which reflect correlation between molecular displacements in perpendicular directions:

$$A = \exp(-b_{xx}D_{xx} - b_{yy}D_{yy} - b_{zz}D_{zz} - 2b_{xy}D_{xy} - 2b_{xz}D_{xz} - 2b_{yz}D_{yz}).$$

Hence, it is important to note that by using diffusion-encoding gradient pulses along one direction only, signal attenuation not only depends on the diffusion

effects along this direction but may also include contribution from other directions.

Now, calculation of the $\mathbf{b}$ -tensor may quickly become complicated when many gradient pulses are used, but the full determination of the diffusion tensor $\mathbf{D}$ is necessary if one wants to assess properly and fully all anisotropic diffusion effects.

To determine the diffusion tensor $\mathbf{D}$ fully, one must first collect diffusion-weighted images along several gradient directions, using diffusion-sensitized MRI pulse sequences such as *echoplanar imaging* (EPI) [Bih95]. As the diffusion tensor is symmetric, measurements along only 6 directions are mandatory (instead of 9), along with an image acquired without diffusion weighting ($b = 0$).

In the case of axial symmetry, only four directions are necessary (tetrahedral encoding), as suggested in the *spinal cord* [CWM00]. The acquisition time and the number of images to process are then reduced.

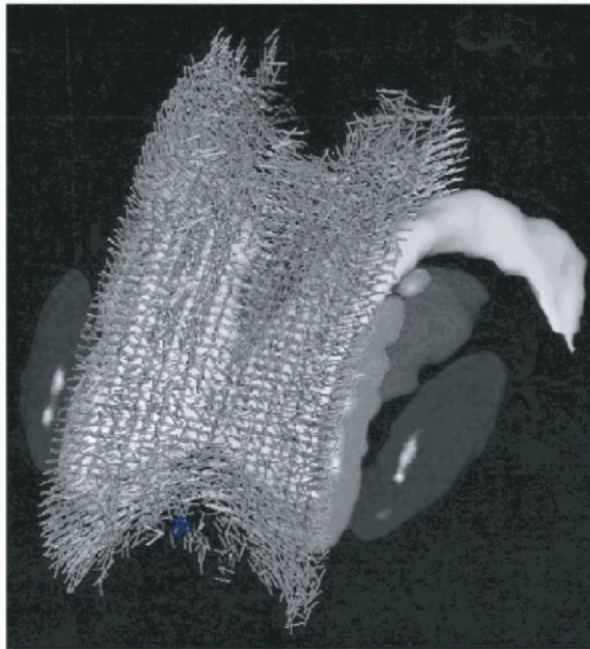


Fig. 1.18. 3D display of the diffusion tensor $\mathbf{D}$. Main eigen-vectors are shown as cylinders, the length of which is scaled with the degree of anisotropy. Corpus callosum fibers are displayed around ventricles, thalami, putamen, and caudate nuclei (adapted from [BMP01]).

In this way, with *diffusion tensor imaging* (DTI), diffusion is no longer described by a single diffusion coefficient, but by an array of 9 coefficients

(dependent on the sampling discrete time) that fully characterize how diffusion in space varies according to direction. Hence, diffusion anisotropy effects can be fully extracted and exploited, providing even more exquisite details on tissue microstructure.

With DTI, diffusion data can be analyzed in three ways to provide information on tissue microstructure and architecture for each voxel: (i) the *mean diffusivity*, which characterizes the overall mean-squared displacement of molecules and the overall presence of obstacles to diffusion; (ii) the *degree of anisotropy*, which describes how much molecular displacements vary in space and is related to the presence and coherence of oriented structures; (iii) the *main direction of diffusivities* (main ellipsoid axes), which is linked to the orientation in space of the structures.

Mean Diffusivity

To get an overall evaluation of the diffusion in a voxel or 3D-region, one must avoid anisotropic diffusion effects and limit the result to an invariant, i.e., a quantity that is independent of the orientation of the reference frame [BMB94]. Among several combinations of the tensor elements, the trace of the diffusion tensor, $\text{Tr}(\mathbf{D}) = D_{xx} + D_{yy} + D_{zz}$, is such an invariant. The mean diffusivity is then given by $\text{Tr}(\mathbf{D})/3$.

Diffusion Anisotropy Indices

Several scalar indices have been proposed to characterize diffusion anisotropy. Initially, simple indices calculated from diffusion-weighted images, or ADCs, obtained in perpendicular directions were used, such as ADC_x/ADC_y and displayed using a color scale [DTP91]. Other groups have devised indices mixing measurements along x , y , and z directions, such as $\frac{\max[ADC_x, ADC_y, ADC_z]}{\min[ADC_x, ADC_y, ADC_z]}$, or the standard deviation of ADC_x , ADC_y , and ADC_z divided by their mean value [GVP94]. Unfortunately, none of these indices are really quantitative, as they do not correspond to a single meaningful physical parameter and, more importantly, are clearly dependent on the choice of directions made for the measurements. The degree of anisotropy would then vary according to the respective orientation of the gradient hardware and the tissue frames of reference and would generally be underestimated. Here again, invariant indices must be found to avoid such biases and provide an objective, intrinsic structural information [BP96].

Invariant indices are thus made of combinations of the eigen-values λ_1 , λ_2 , and λ_3 of the diffusion tensor $\mathbf{D}$ (see Figure 1.18). The most commonly used invariant indices are the *relative anisotropy* (RA), the *fractional anisotropy* (FA), and the *volume ratio* (VR).

Fiber Orientation Mapping

The last family of parameters that can be extracted from the DTI concept relates to the mapping of the orientation in space of tissue structure. The

assumption is that the direction of the fibers is collinear with the direction of the eigen-vector associated with the *largest eigen-diffusivity*. This approach opens a completely new way to gain direct and in vivo information on the organization in space of oriented tissues, such as muscle, myocardium, and brain or spine white matter, which is of considerable interest, both clinically and functionally. Direction orientation can be derived from DTI directly from diffusion/orientation-weighted images or through the calculation of the diffusion tensor $\mathbf{D}$. Here, a first issue is to display fiber orientation on a voxel-by-voxel basis. The use of color maps has first been suggested, followed by representation of ellipsoids, octahedra, or vectors pointing in the fiber direction [BMP01].

Brain Connectivity Studies

Studies of neuronal connectivity are important to interpret functional MRI data and establish the networks underlying cognitive processes. Basic DTI provides a means to determine the overall orientation of white matter bundles in each voxel, assuming that only one direction is present or predominant in each voxel, and that diffusivity is the highest along this direction. 3D vector-field maps representing fiber orientation in each voxel can then be obtained back from the image data through the diagonalization (a mathematical operation which provides orthogonal directions coinciding with the main diffusion directions) of the diffusion tensor determined in each voxel. A second step after this *inverse problem* is solved consists in *connecting* subsequent voxels on the basis of their respective fibre orientation to infer some continuity in the fibers (see Figure 1.19). Several algorithms have been proposed. *Line propagation algorithms* reconstruct tracts from voxel to voxel from a seed point. Another approach is based on *regional energy minimization* (minimal bending) to select the most likely trajectory among several possible [BMP01, BZ02, Bih03].

1.2 Cognition and Associative Memory

Recall that the term *cognition* (Latin: *cognoscere* = *to know*) is used in several loosely-related ways to refer to a facility for the human like processing of information, applying knowledge and changing preferences.²

² For example, according to the so-called *top-down object-based goal-oriented approach* (TOGA) standard, the Information-Preferences-Knowledge *cognitive architecture* consists of:

Data: everything what is/can be processed/transformed in computational and mental processes. Concept data is included in the ontology of ‘elaborators’, such as developers of methods, programmers and other computation service people. In this sense, data is a relative term and exists only in the couple (data, processing).

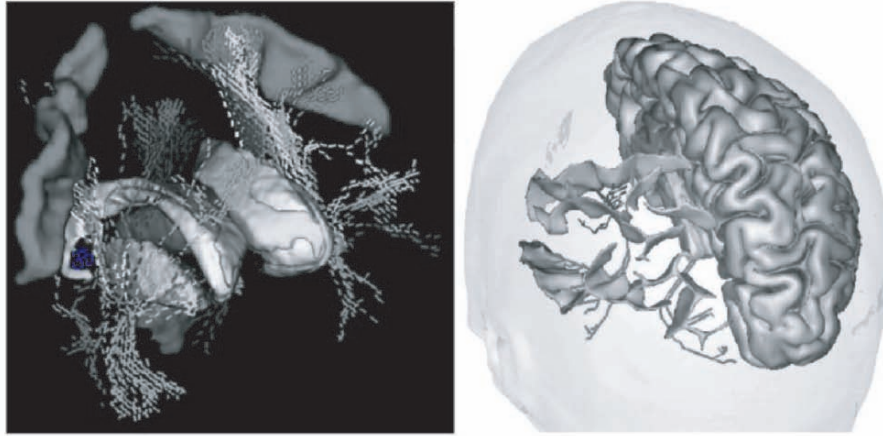


Fig. 1.19. Several approaches have been developed to ‘connect’ voxels after white matter fibers have been identified and their orientation determined. Left: 3D display of the motor cortex, central structures and connections. Right: 3D display from MRI of a brain hemisphere showing sulci and connections.

Information: data which represent a specific property of the domain of human or artificial agent’s activity (such as: addresses, tel. numbers, encyclopedic data, various lists of names and results of measurements). Every information has always a source domain. It is a relative concept. Information is a concept from the ontology of modeler/problem-solver/decision-maker.

Knowledge: every abstract property of human/artificial agent which has ability to process/transform a quantitative/qualitative information into other information, or into another knowledge. It includes: instructions, emergency procedures, exploitation/user manuals, scientific materials, models and theories. Every knowledge has its reference domain where it is applicable. It has to include the source domain of the processed information. It is a relative concept.

Preference: an ordered relation among two properties of the domain of activity of a *cognitive agent*, it indicates a property with higher utility. Preference relations serve to establish an intervention goal of an agent. Cognitive preferences are relative. A preference agent which manages preferences of an intelligent agent can be external or its internal part.

Goal: a hypothetical state of the domain of activity which has maximal utility in a current situation. Goal serves to the choice and activate proper knowledge which process new information.

Document: a passive carrier of knowledge, information and/or preferences (with different structures), comprehensive for humans, and it has to be recognized as valid and useful by one or more human organizations, it can be physical or electronic.

Computer Program: (i) from the modelers and decision-makers perspective: an active carrier of different structures of knowledge expressed in computer languages and usually focused on the realization of predefined objectives (a design-goal). It may include build-in preferences and information and/or request specific IPK as

Cognition, (consisting of a set of cognitive processes) can be natural or artificial, conscious or subconscious. Therefore, it can be analyzed from different perspectives and in different contexts, in neurology, psychology, philosophy, system science and computer science. The concept of cognition is closely related to such abstract concepts as, mind, reasoning, perception, intelligence, learning, and many others, which describe numerous capabilities of human mind and expected properties of artificial, or synthetic intelligence. Cognition is an abstract property of advanced living organisms, so it is studied as a direct property of a brain, or of an abstract mind on symbolic and sub-symbolic levels.

In psychology and AI, it is used to refer to the mental functions, mental processes and states of intelligent entities (humans, human organizations, highly autonomous robots), with a particular focus toward the study of such mental processes as, comprehension, inferencing, decision-making, planning and learning (see also cognitive science and cognitivism). Recently, advanced cognitive researchers are especially focused on the capacities of abstraction, generalization, concretization/specialization and meta-reasoning which descriptions involve such concepts as, beliefs, knowledge, desires, preferences and intentions of intelligent individuals/objects/agents/systems.

The term ‘cognition’ is also used in a wider sense to mean the act of knowing or knowledge, and may be interpreted in a social or cultural sense to describe the emergent development of knowledge and concepts within a group that culminate in both thought and action.

On the other hand, recall that Teuvo Kohonen’s *associative memory* (AM), also called the *content-addressed memory*³ [Koh82, Koh88], or *associative storage*, or *associative array*, represents a high-speed-searching memory organization in which the *memory is accessed by its content*, which is opposed to an explicit address by the memory location within the certain storage medium. Thus, reference clues are ‘associated’ with actual memory contents until a desirable match (or set of matches) is found. For example, AM stands as the most likely model for various cognitive memories. At the same time, obvious examples of systems that employ such a memory are *production systems*, tools used by AI (especially within the *expert systems*), consisting of:

1. A database of rules;
2. A working memory;
3. A pattern matcher; and
4. A procedure that resolves conflicts between the rules.

Naturally, humans retrieve information best when it can be *associated* (or linked) to other related information. This *association-map*, or more generally,

data. (ii) from the software engineers perspective: a data-processing tool (more precise technical def. you may find on the Web).

³ Note that conventional shortcut is CAM, not AM, which we use for simplicity.

association-functor (see below) is fast, direct, many-to-many, and homomorphic.

In modern computer technology (see, e.g., [KW97]), unlike standard computer memory (random access memory, or RAM) in which the user supplies a memory address and the RAM returns the data word stored at that address, an AM is designed such that the user supplies a data word and the AM searches its entire memory to see if that data word is stored anywhere in it. If the data word is found, the AM returns a list of one or more storage addresses where the word was found (and in some architectures, it also returns the data word, or other associated pieces of data). Thus, an AM is the hardware embodiment of what in software terms would be called an associative array. As an AM is designed to search its entire memory in a single operation, it is much faster than RAM in virtually all search applications.⁴

AM is often used in *computer networking devices*. For example, when a network switch receives a packet from one of its ports, it updates an internal table with the packet's source *MAC address*⁵ and the port it was received on. It then looks up the destination MAC address in the table to determine what port the packet needs to be forwarded to, and sends it out that port. The MAC address table is usually implemented with a binary AM so the destination port can be found very quickly, reducing the switch's latency.

⁴ Note, however, that there are cost disadvantages to AM. Unlike a RAM chip, which has simple storage cells, each individual memory bit in a fully parallel AM must have its own associated comparison circuit to detect a match between the stored bit and the input bit. Additionally, match outputs from each cell in the data word must be combined to yield a complete data word match signal. The additional circuitry increases the physical size of the AM chip which increases manufacturing cost. The extra circuitry also increases power dissipation since every comparison circuit is active on every clock cycle. Consequently, AM is only used in specialized applications where searching speed cannot be accomplished using a less costly method. To achieve a different balance between speed, memory size and cost, some implementations emulate the function of AM by implementing standard tree search or hashing designs in hardware, using hardware tricks like replication or pipelining to speed up effective performance. These designs are often used in routers.

⁵ In computer networking a Media Access Control address (MAC address) is a unique identifier attached to most forms of networking equipment. Most layer 2 network protocols use one of three numbering spaces managed by the IEEE: MAC-48, EUI-48, and EUI-64, which are designed to be globally unique. Not all communications protocols use MAC addresses, and not all protocols which do require such globally unique identifiers. The IEEE claims trademarks on the names 'EUI-48' and 'EUI-64'. ARP/RARP is commonly used to map the layer 2 MAC address to an address in a layer 3 protocol such as Internet Protocol (IP). On broadcast networks such as Ethernet the MAC address allows each host to be uniquely identified and allows frames to be marked for specific hosts. It thus forms the basis of most of the layer 2 networking upon which higher OSI Layer protocols are built to produce complex, functioning networks.

Ternary AMs are used in network routers, where each address has two parts: the network address, which can vary in size depending on the *sub-net configuration*, and the host address, which occupies the remaining bits.⁶

In software engineering, an *associative array* (also called *dictionary*, *finite map*, *lookup table*,⁷ and in query-processing an *index file*) is an *abstract data type*⁸ composed of a collection of keys and a collection of values, where each key is associated with one value. The operation of finding the value associated with a key is called a lookup or indexing, and this is the most important operation supported by an associative array. The relationship between a key and its value is sometimes called a *mapping* or *binding*. Associative arrays are very closely related to the mathematical concept of a *function with a finite domain*. As a consequence, a common and important use of associative arrays is in *memoization*.⁹ From the perspective of a programmer using an

⁶ Each sub-net has a network mask that specifies which bits of the address are the network address and which bits are the host address. Routing is done by consulting a routing table maintained by the router which contains each known destination network address, the associated network mask, and the information needed to route packets to that destination. Without AM, the router compares the destination address of the packet to be routed with each entry in the routing table, performing a logical AND with the network mask and comparing it with the network address. If they are equal, the corresponding routing information is used to forward the packet. Using a ternary AM for the routing table makes the lookup process very efficient. The addresses are stored using ‘don’t care’ for the host part of the address, so looking up the destination address in the AM immediately retrieves the correct routing entry; both the masking and comparison are done by the AM-hardware.

⁷ Recall that in computer science, a lookup table is a data structure, usually an array or associative array, used to replace a runtime computation with a simpler lookup operation. The speed gain can be significant, since retrieving a value from memory is often faster than undergoing an expensive computation.

⁸ Recall that in computing, an abstract data type (ADT) is a specification of a set of data and the set of operations that can be performed on the data. Such a data type is abstract in the sense that it is independent of various concrete implementations. The definition can be mathematical, or it can be programmed as an interface. The interface provides a constructor, which returns an abstract handle to new data, and several operations, which are functions accepting the abstract handle as an argument.

⁹ Recall that Memoization is a technique used to speed up computer programs by storing the results of functions for later reuse, rather than recomputing them. Memoization is a characteristic of *dynamic programming*. Functions can only be memoized if they are referentially transparent – that is, if they will always return the same result given the same arguments. Operations which are not referentially transparent, but whose results are not likely to change rapidly, can still be cached with methods more complicated than memoization. In general, memoized results are not expired or invalidated later, while caches generally are. In imperative languages, both memoization and more general caching are typically implemented using some form of associative array. In a *functional programming*

associative array, it can be viewed as a *generalization of an array*: While a regular array maps integers to arbitrarily typed objects, an associative array maps arbitrarily typed objects to arbitrarily typed objects. The operations that are usually defined for an associative array are:

- (i) Add: Bind a new key to a new value;
 - (ii) Reassign: Bind an old key to a new value;
 - (iii) Remove: Unbind a key from a value and remove it from the key set;
- and
- (iv) Lookup: Find the value (if any) that is bound to a key.

All these machines, most notably *artificial neural networks* (ANNs, see section below), have capability of storing, recognizing and classifying *spatio-temporal patterns*. A *pattern* can be stored in ANN by associating (through an appropriate choice of the connections among the neurons of a network) the configurations of the nervous activity elicited by the presentation of the pattern itself with a steady-state, or *attractor*, of the network. Many physical systems can operate as AMs, once their dynamical evolutions converge towards their attractors (from arbitrary initial conditions).

Recall that above mentioned *attractor* is a set of system's states (i.e., points in the system's phase-space), invariant under the dynamics, towards which neighboring states in a given *basin of attraction* asymptotically approach in the course of dynamic evolution.¹⁰ An attractor is defined as the smallest unit which cannot be itself decomposed into two or more attractors with distinct basins of attraction. This restriction is necessary since a dynamical system may have multiple attractors, each with its own basin of attraction. Conservative systems do not have attractors, since the motion is periodic. For dissipative dynamical systems, however, volumes shrink exponentially, so attractors have 0 volume in nD phase-space. In particular, a stable *fixed-point* surrounded by a dissipative region is an attractor known as a *map sink*.¹¹ Regular attractors (corresponding to 0 *Lyapunov exponents*) act as *limit cycles*, in which trajectories circle around a limiting trajectory which they asymptotically approach, but never reach. The so-called *strange attractors*¹² are bounded regions of phase-space (corresponding to positive Lyapunov characteristic exponents) having zero measure in the embedding

language it is possible to construct a higher-order function memoize which will create a memoized function for any referentially transparent function. In languages without higher-order functions, memoization must be implemented separately in each function that is to benefit from it.

¹⁰ A *basin of attraction* is a set of points in the system's phase-space, such that initial conditions chosen in this set dynamically evolve to a particular attractor.

¹¹ A *map sink* is a stable fixed-point of a map which, in a dissipative dynamical system, is an attractor.

¹² A strange attractor is an attracting set that has zero measure in the embedding phase-space and has fractal dimension. Trajectories within a strange attractor appear to skip around randomly.

phase-space and a *fractal dimension*. Trajectories within a strange attractor appear to skip around randomly.

In psychology, there is an AM-related term *trans-derivational search* (TDS), meaning when a search is being conducted for a fuzzy match across a broad field. In computing the equivalent function can be performed using AM. Unlike usual searches, which look for literal (i.e., exact, logical, or regular expression) matches, a trans-derivational search is a search for a possible meaning or possible match as part of communication, and without which an incoming communication cannot be made any sense of whatsoever. It is thus an integral part of processing language, and of attaching meaning to communication. An example of TDS use is in *Ericksonian hypnotherapy*,¹³ where vague suggestions are used that the patient must process intensely to find their own meanings for, thus ensuring that the practitioner does not intrude his own beliefs into the subject's inner world.

1.3 Categorical Language for Neuro-Cognition

From cognitive science perspective, human brain is an infinitely-complex *association machine* with five major information-processing functions (see [Buz91]):

1. *Input* – any information taken into the brain by any of the senses;
2. *Storage/Recall* – including both short-term (or, working) memory and long-term memory;
3. *Analysis* – pattern recognition/classification;
4. *Output* – any form of communication or creative act, including thinking;
5. *Control* – feedback control of all mental and physical functions.

Now, recall from Appendix that theory of *categories and functors* is a synthetic metalanguage of modern mathematics, visually depicted using commutative diagrams. In this section we show that categories and functors form the natural framework for *hierarchical associative memory organization*.

On the other hand, mathematical models for complex systems, such as neural, biological or social systems, have been given by many authors (see

¹³ Hypnotherapy is the application of hypnosis as a form of treatment, usually for relieving pain or conditions related to one's state of mind. Practitioners believe that when a client enters, or believes he has entered, a state of trance, the patient is more receptive to suggestion and other therapy. The most common use of hypnotherapy is to remedy maladies like obesity, smoking, pain, ego, anxiety, stress, amnesia, phobias, and performance but many others can also be treated by hypnosis, including functional disorders like Irritable Bowel Syndrome. American psychiatrist M. H. Erickson is noted for his often unconventional approach to psychotherapy, such as described in the book 'Uncommon Therapy' by Jay Haley and the book 'Hypnotherapy: An Exploratory Casebook' by Milton H. Erickson and Ernest L. Rossi. New York, Irvington Publishers, Inc. (1979)

[II06b]), often in the general framework of ‘relational biology’ (see [II06a]). This framework was defined in four major steps (followed by a number of authors, including M. Arbib [Arb66, Arb87]):

- Rosen’s [Ros58a] theory of (M, R) –systems,
- Rashevsky’s [Ras67] theory of *organismic* systems, and
- Louie’s [Lou83] theory of categorical systems *interacting with measuring* devices.
- A. Ehresmann’s and Vanbremeersch’s [EV87, EV96] fully developed (an ordinary) categorical theory of *natural* systems, which we present in this subsection in a form adapted for brain dynamics.

1.3.1 Categorical Patterns and Hierarchical Links

In a category, a *pattern* P is modeled by the data of a family of objects N_i and of some distinguished links between them [EV87, EV96]. A *collective link* from the pattern P to another object N' is a family of individual links f_i from each N_i to N' , correlated by the distinguished links of the pattern (see Figures 1.20 and 1.21 below).

The object binding the pattern (if it exists) in a category is modeled by the *colimit* (or *inductive limit*) [Kan88, MacL71] of the pattern. An object N is the colimit, or the binding, of the pattern P if it satisfies the two conditions:

1. There exists a collective link (c_i) from the pattern to N ,
2. Each collective link (f_i) from the pattern to any object N' binds into a unique arrow f from N to N' .

If a pattern has a colimit, it is unique (up to an isomorphism).

The situation can be seen ‘upside-down’: if an object N is the colimit of a pattern P , we say that N is a complex object admitting P as a *decomposition*. An object may have different decompositions. Indeed, while a pattern has at most one colimit (up to an isomorphism), several patterns may have the same colimit [EV87, EV96].

Limit (or *projective limit*) of a pattern P in a category actualizes the capacity of objects to collectively decode and classify information of which each receives only some part (see Figure 1.20).

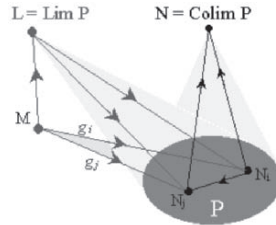


Fig. 1.20. A *limit* and a *colimit* of a *pattern* in a category.

A common message from an object M of the category to a pattern P is a family of individual links g_i from M to the N_i , which are correlated by the distinguished links of P . The limit of P , if it exists, is an object L which ‘classifies’ the common messages in the sense that the family (g_i) uniquely factors into an arrow g from M to L .

Formally, the *limit* is defined as the *colimit* of the pattern in the *dual* (opposite category).

In brain-like systems, the objects are partitioned into different complexity levels, each level satisfying its own rules. There are intra-level brain links, but also inter-level brain links. An object of brain level $n + 1$ is an aggregate of objects of brain level n , bound by strong interactions between them which generate their cohesion in the brain aggregate.

Such a hierarchical brain-like organization is modeled in a category using the *colimit* operation [Kan88, MacL71]. A brain aggregate N can be modeled by the colimit of a brain pattern of linked objects representing its internal organization [EV87, EV96].

A *pattern* P in a *brain category* $\mathcal{B}$ is defined as a family of objects N_i of $\mathcal{B}$ and some distinguished links between them [EV87, EV96]. A *collective link* from P to an object N' is a family of arrows $(f_i : N_i \rightarrow N')$ correlated by the distinguished arrows of the pattern. Collective links model coherent brain interactions (constraints, energy, or information transfer) effected by all the N_i acting in cooperation along their distinguished links, which could not be realized by the objects of the pattern acting individually (see Figure 1.21).

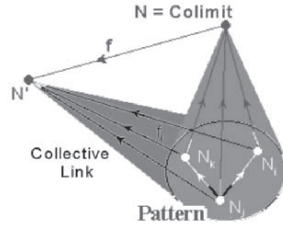


Fig. 1.21. A *pattern* in a category, a *collective link*, and a *colimit*.

A pattern $P \in \mathcal{B}$ may have collective links toward several brain objects. P has an aggregate if, among those collective links, there is an ‘optimal’ one, i.e., a $(c_i : N_i \rightarrow N)$ through which all the others factorize. In categorical terms, this means that N is the colimit of P ; we also say that N binds P , or, seen in the opposite direction, that P is a *decomposition* of the complex object N . The colimit has localized properties in being an objective representation of the brain pattern it binds, that so acquires a dynamical stability; but it has also a ‘global’ implication for its surround through its ‘universal’ property of factorizing every collective link [EV87, EV96].

The brain's colimit actualizes the potentiality of the objects to act together in a coherent manner by integrating the pattern in a higher unit. In a natural system where the links have a given 'strength', the formation of a colimit is characterized in two ways:

1. 'Locally and structurally', a strengthening of the distinguished brain links of the brain pattern restricts the DOF of the objects to ensure a more efficient cooperation among them;
2. 'Universally and functionally', the actions of the brain's colimit on the other objects of the brain system subsume the activity of the whole pattern (they correspond to its collective links).

The brain's colimit 'forgets' the precise organization of the brain pattern and records only its collective actions, and these can be the same for more or less differing brain patterns.

1.3.2 A General Natural System

The state of a *general natural system*, such as a neural, biological, or social system, at a given time t can be modeled by a category: its objects represent the components (of any level) of the system, and the morphisms their links in the system at this time. These links may represent more or less stable structural links such as causal or topological relations (e.g., desmosomes between adjacent cells), channels transmitting information, spatial or energetic constraints, or temporary interactions between two components (see [EV87, EV96]).

The composite of two successive links $N \rightarrow N'$ and $N' \rightarrow N''$ represents the cumulative interaction from N to N'' obtained by taking first the interaction from N to N' , then the interaction from N' to N'' .

Thus an object N plays a double role:

1. It acts as an agent, or an emitter, through its links toward the other objects which represent its actions, or the messages it sends, and
2. It becomes a receptor, or an observer, through the links toward it, which correspond to the aspects it observes, or the information it receives, or the constraints which are imposed to it.

In a biological system, the identity of an object models its 'self'. The composition of links determines classes of paths which are 'functionally equivalent' (i.e., with the same practical effect), namely all the paths between two objects having the same composite; it allows to characterize the various temporally and energetically equivalent paths by which an information can be transmitted from a component to another one.

The system may have had some components before the time t which no more figure in it at t . These 'lost' components can be modeled by adding to the category a zero object (or initial object) 0 , i.e., an object such that there exists a unique link from 0 to any other object. We say that we have a category with a zero-object.

Categories modeling natural systems are often constructed in the following way [EV87, EV96]. The graph G of generators is a *labeled graph* in the sense that a real number (or a vector), called its *arrow weight*, is associated to each arrow; this weight represents the force and/or the propagation delay with which information are transmitted by the arrow. A path of G is weighted by the sum of the weights of its factors; and the relations identify two paths c and c' which have the same source, target and weight. The deduced category $\mathcal{K}$ has for links from A to B classes of paths from A to B which have the same weight. It is a *weighted category* in the following sense: A category is weighted if it is a labeled graph and if the weight of a composite $g \circ f$ is the sum of the weights of f and g .

1.3.3 The Category of Neurons

The *category of neurons* $\mathcal{N}$ models a neuronal system at a given time t . It is constructed by the data of *generators* and *relations* [EV87, EV96].

First we define the labeled graph G of its generators. Its vertices are the neurons; the activity of a neuron N at t is determined by its instantaneous frequency of spikes. An arrow from N to N' corresponds to a synapse such that N is the presynaptic neuron and N' the post-synaptic neuron. Let us remark that two neurons may be linked by one, several or no arrows. The weight of an arrow from N to N' represents the strength of the synapse, related to the probability that the activity of N be propagated to N' and to the delay of propagation. We adopt the usual convention that the activity of a neuron N is the summation of the activities of the neurons N_i to which it is linked, pondered by the weights of the synapses linking these N_i to N .

The category of paths $\mathcal{C}$ on this graph admits the neurons for its objects, the links from N to N' are the paths formed by successive synapses, and the composition of paths is their concatenation. In this category, two paths from N to N' have the same weight if the activity of N is transmitted to N' in the same way along one of the other of the paths.

The category $\mathcal{N}$ is deduced from the data of G and of the relations on $\mathcal{C}(G)$ identifying two paths from N to N' which have the same weight. Thus its objects are still the neurons, but the links from N to N' , called *multisynaptic paths*, are classes of paths with the same weight. It is a weighted category.

1.3.4 Memory Evolutive System

Memory evolutive system (MES, for short [EV87, EV96]) represents a categorical model to study biological, neural or social systems and to give plausible answers to several questions. The first question is how to model ‘open’ systems, whose components, with various complexity levels, change in time through exchanges with the environment and through internal modifications such as the formation of new complex objects? This question has led to introduce the notion of an *evolutive system* (ES); in it the state of the system

at a given time, with its components and their interactions, is modeled by a category; a ‘complex’ object is represented by the colimit (or inductive limit) of a pattern of linked objects standing for its internal organization; and the changes between successive states by ‘transition’ functors between the corresponding state categories. The ES is hierarchical if its objects are partitioned into complexity levels, so that an object of level $n + 1$ is the colimit of a pattern of linked objects of level n . In an ES, the ‘archetypal’ changes (like birth/death) are modeled by a complexification process with respect to a ‘strategy’ of which the objectives are to add or suppress objects (external exchanges), and to form complex objects (adjunction of a colimit to a pattern) or to decompose complex objects.

This raises the problem of selection of strategies in an autonomous system. Whence the introduction of the notion of a partial internal *center of regulation* (CR), able to develop a stepwise ‘trial-and-error’ learning process, thanks to the formation, at each step, of its own internal (more or less deformed) representation of the system, its landscape, on which a strategy is selected.

The case of a neural system shows that its evolution depends on a whole net of competitive CRs. It has led to the definition of *memory evolutive systems* (MES): a MES is an ES with a net of cooperating/competitive CRs, which develop a hierarchical sub-ES, called *memory*, which allows to record past experiences in a flexible manner for a better later adaptation; each CR has its differential access to this memory. The links between objects have a ‘strength’ and a ‘propagation delay’ (represented by the set $\mathbb{R}$ of real numbers).

In a MES, each CR_i acts on its landscape according to its own timescale, but the strategies of the different CRs compete for the common resources of the system, and at each time there is a ‘play’ among these possibly conflicting strategies, with a risk of fractures for some CRs; whence a dialectics between CRs which are heterogeneous by their complexity level and their timescale, generating the complexity of the system. Time has an essential role through the structural temporal constraints imposed on each CR, and repair processes develop for their being respected anew if they are transgressed. In particular, this leads to an aging theory for an organism based on a cascade of de/re-synchronizations for more and more complex CRs.

The problem of reductionism can be studied in the framework of MES: an object of level $n + 1$ is always the $(n - k)$ -iterated colimit of a ramification [EV87, EV96], based on the level $k < n$, but is it also ‘reducible’ to the level k in only one step, i.e., is it a simple colimit of a pattern of level k ? A more fine analysis of the form of the links between objects led to a characterization of such reducible objects, and proved that there exist non-reducible objects if the MES satisfies the multiplicity principle: there exist non-equivalent patterns which have the same colimit. In this case, a sequence of complexifications cannot be replaced by a unique complexification with respect to a strategy subsuming the intermediate strategies.

The preceding results applied to a neural system allow to describe an ‘algebra of mental objects’, and they explain the development of a procedural

memory, and the formation of higher order cognitive processes. They suggest an approach to the mind/body problem: mental states are not simple physical states of the brain (activation of synchronous assemblies of neurons), but they dynamically supervene on such states through the progressive unfolding of a ramification based on the level of those physical states.

1.3.5 Neural System in a Category

Assemblies of Neurons

The response of a neuronal system to a simple signal is the activation of a specialized neuron; e.g., in the visual areas, there exist neurons activated by a segment of a certain direction ('simple' cells), or by an angle ('complex' cells),... [EV87, EV96]. But more complex signals, apart from a few exceptions (e.g., a neuron representing a hand holding a banana in the monkey), do not have their own 'grandmother neuron'. The development of neuronal imagery shows that complex signals, or motor programs, are represented by the short synchronization of a specific assembly of neurons. And learning would consist in the formation of such synchronous assemblies, under the effect of the reinforcement of the synapses between their neurons, while following the Hebbian rule [Heb49]: a synapse between two neurons is strengthened if the two neurons are active at the same time, and its force decreases if one is active while the other is not.

These processes can be described in a MES modeling a neuronal system, and this model makes it possible to understand how they form the basis for the development of increasingly complex mental objects, enjoying a great flexibility.

Memory Evolutive System of Neurons

In such a MES, the state-categories will be obtained by successive complexifications of a category of neurons defined as follows [EV87, EV96]:

Consider the graph whose vertices are the neurons at the time t , and the arrows f from N to N' are the synapses having N and N' respectively as their presynaptic and post-synaptic neurons. The category of paths of this graph is weighted, the weight of a synaptic path being connected to the probability that the activity of N (determined by its instantaneous frequency of impulses) is propagated to N' , and to the propagation delay of this activation. The category of neurons at t is obtained from this category by identifying two synaptic paths from N to N' with the same weight.

An assembly of neurons is represented by a pattern P in such a category of neurons. Its synchronization is then modeled by the emergence of a colimit of P in a complexification of the category; this colimit operates like a single 'neuron of a higher order' integrating the assembly, and which takes its own identity; it is called a neuron of category (or cat-neuron). The construction of the complexification determines which are 'the good' links between

cat-neurons (namely the simple and complex links), therefore between synchronous assemblies of neurons. And, by iteration of the process of complexification, one can define cat-neurons of order 2 representative of assemblies of assemblies of neurons, then of order 3 etc..., modeling increasingly complex mental objects, or cognitive processes of a higher order. That makes it possible to describe explicitly how an *algebra of mental objects* develops [EV87, EV96]. In particular, extension of the memory, under the effect of different CRs, leads to the emergence of cat-neurons of increasing orders. Among them, the cat-neurons which ‘memorize’ the strategies carried out and their results form the procedural memory.

Semantics

The neuronal system of a higher animal will be able to classify the items it recognizes by the formation of classes of invariance [EV87, EV96].

For a lower CR, say E , this classification will be only ‘pragmatic’: two items ‘are acted’ as equivalent when their traces in the landscape activate the same pattern of agents: e.g., it is the same pattern of agents of a ‘color’ CR which is activated by all the blue objects.

But this classification will take a ‘meaning’ only at the level of a higher CR, with a longer period, which can determine what have in common the different items that E has classified together, and memorize such a class in the form of an object called an E -concept. This object will be modeled by the limit of the pattern of agents of E activated by all the items of the class, and its various instances form the class of invariance of the concept (e.g., for the blue color-concept, all representations of blue objects).

The CR-concepts relating to the different CRs will form a Semantic Memory which is developed under the effect of successive complexifications, by addition of concepts classifying several attributes (such as a ‘blue triangle’), then of more abstract concepts obtained as limits of patterns of such ‘concrete’ concepts, linked by complex links. A concept can be seen as a prototype for a class of items having a ‘family air’ (in the sense of Wittgenstein), and it does not suppose the existence of a language. The activation of a concept rests on a double indetermination: initially choice of a particular instance of the concept, then choice of a particular decomposition of this instance. It results from it that the introduction of a semantics makes the interplay among the strategies of the different CRs still more flexible; indeed the procedural memory will become semantic, so that, in higher CRs, the choice of a strategy can be made in the form of a concept, without specifying a particular object of the class of invariance of the concept. This gives a new DOF in the formation of the effective strategy on the system, since it makes it possible to activate the instance best adapted taking account of the strategies reflected by other CRs. For example the command for lifting an object will be modulated according to the shape and the size of the object to lift.

During the evolution of a MES, regrouping of emerging components and links can lead to the formation of new CRs of increasingly high levels, devel-

oping particular aptitudes. In particular, a MES equipped with a semantic memory can develop ‘conscious’ CRs, able to have an internal view of the semantics and of the concept of time. Such a CR can be characterized by the following capacities [EV87, EV96]:

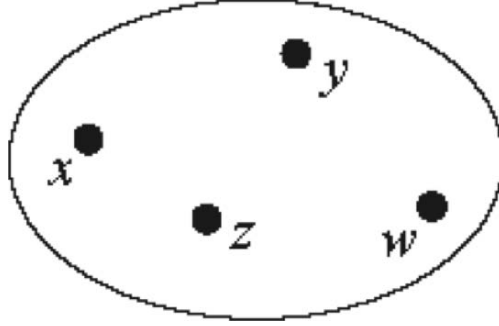
1. Retrospectively to extend its current landscape to lower levels of the recent past by an increase in attention, in particular after a fracture;
2. To operate a process of abduction in this extended landscape to find possible causes of the fracture; and
3. To plan a choice of strategies covering several future steps by the formation of ‘virtual’ landscapes in which the strategies (selected in the form of concepts) can be tested without energy cost.

From the neurological point of view, these properties rest on the existence of functional loops between various areas of the brain.

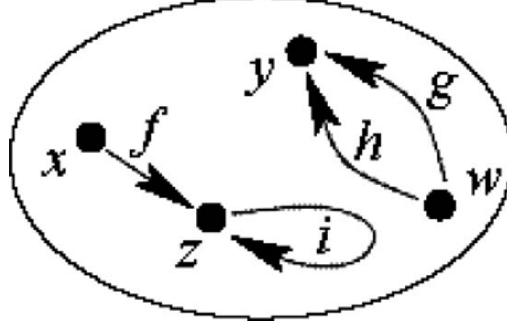
1.4 Hierarchical Associative Machinery of n –Functors

In this final section of the Introduction, we present our *global associative machinery for brain and cognition modeling*. In the subsequent Chapters we will give the particular cases of this global machinery.

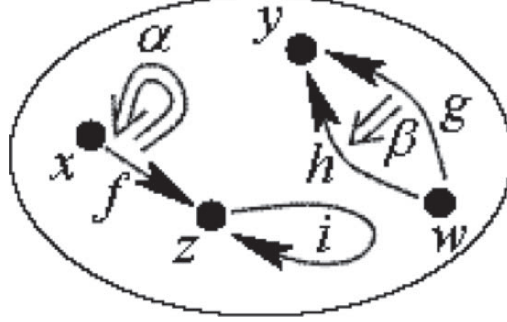
Recall (from Appendix) that in modern *global modeling of arbitrary dynamical systems* (or *processes*) by means of n –categories, instead of classical mathematical description based on a chosen *set of things, objects, points*, e.g., $\{x, y, z, w\}$, depicted as:



we can now start our modeling with a *category of things* ($\{x, y, z, w\}$) and *processes between them* (paths/arrows $\{f, i, h, g\}$), depicted as:



or a 2-category of things ($\{x, y, z, w\}$), processes ($\{f, i, h, g\}$), and processes between processes (double arrows $\{\alpha, \beta\}$), depicted as:



... and so on.

Now, recall that a *functor* represents a natural association between two categories. It is a *generic picture*, projecting one category (with all its things and processes between them, etc.) into another category of similar structure (see Appendix). Intuitively, a functor is a picture projecting a ‘film category’ onto a ‘screen category’. Rigorously, a functor is a generalization/abstraction of the concepts of *functions* (as maps between sets) and *operators* (as maps between spaces). A *functor is a natural association modeling tool*. Formally, it is written $F : S \Rightarrow T$, where S is the *source category*, T is the *target category*, and F is the *association functor* between them.

In this way, we can naturally extend the above *structural modeling machinery* of n -categories to develop an associated *functional modeling machinery* of n -functors. Namely, instead of classical modeling of a simple system/process with a function, which is a one-to-one (and ‘onto’) map between two sets (see Figure 1.22 (a)), we can now model a complex system/process with an *ordinary functor* (see Figure 1.22 (b)),¹⁴ or a very com-

¹⁴ Here we have the following processes: $\{f : x \rightarrow z, i : z \rightarrow z, \text{ and } g, h : w \rightarrow y\}$ (left) and $\{F(f) : F(x) \rightarrow F(z), F(i) : F(z) \rightarrow F(z), \text{ and } F(g), F(h) : F(w) \rightarrow F(y)\}$ (right).

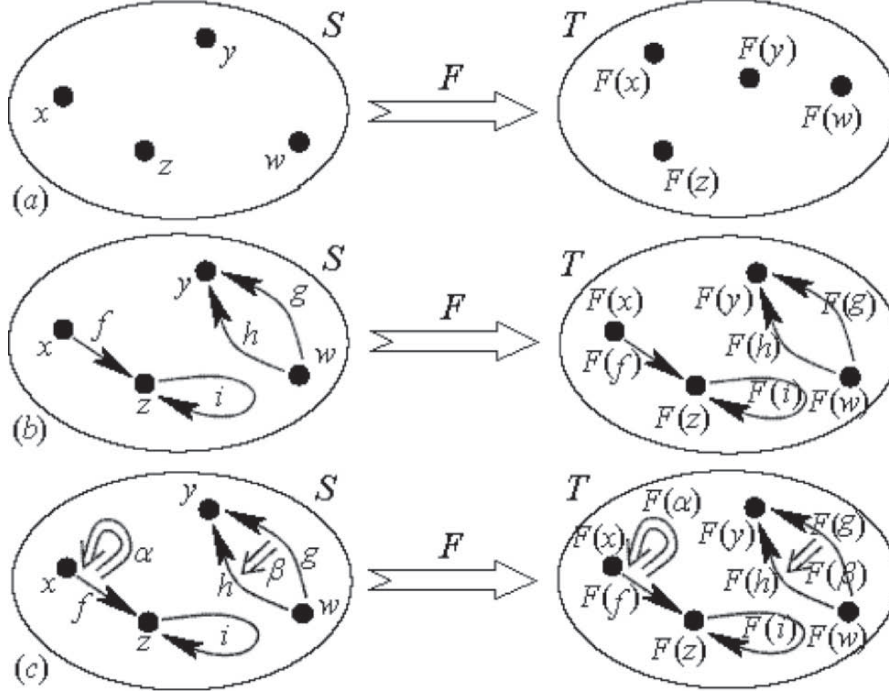


Fig. 1.22. Hierarchical associations modeled as n -functors F : (a) a one-to-one (and ‘onto’) map between two sets, representing a 0-functor; (b) an ordinary functor (or 1-functor) between two ordinary categories; (c) a 2-functor between two 2-categories, etc. In each case, a functor F is projecting a ‘picture’ from some *source* ‘film’ category (with all its things and processes between things etc.) onto the associated *target* ‘screen’ category.

plex system/process with a 2-functor (see Figure 1.22 (c)),¹⁵ ... and so on – recursively.

As a rigorous example of an ordinary 1-functor, recall that the central concept in the realm of *human biodynamics* is the *covariant force law* (see subsection 2.1.7 below, as well as [II05, II06a, II06b]):

$$F_i = mg_{ij}a^j, \quad \text{which ‘in plain English’ reads :}$$

$$\text{Force 1-form-field} = \text{Mass distribution} \times \text{Acceleration vector-field}$$

and the associated commutative diagram, rigorously defining the *force functor* $\mathcal{F}_1$:

¹⁵ Here we have the following processes between processes: $\{\alpha : f \rightarrow f, \beta : g \rightarrow h\}$ (left) and $\{F(\alpha) : F(f) \rightarrow F(f), F(\beta) : F(g) \rightarrow F(h)\}$ (right).

$$\begin{array}{ccc}
TT^*M & \xrightarrow{\mathcal{F}_1} & TTM \\
\uparrow F_i = \dot{p}_i & & \uparrow a^i = \dot{v}^i \\
T^*M = \{x^i, p_i\} & \xleftarrow{L} & TM = \{x^i, v^i\} \\
\swarrow p_i & & \nearrow v^i = \dot{x}^i \\
M = \{x^i\} & &
\end{array} \quad (1.3)$$

which states that the force 1-form, $F_i = \dot{p}_i$, defined on the mixed tangent-cotangent bundle TT^*M , *causes* the acceleration vector-field, $a^i = \dot{v}^i$, defined on the second tangent bundle TTM of the biodynamic configuration manifold M .¹⁶ The left vertical branch (or, left 1-cell TT^*M) of the force functor $\mathcal{F}_1$ represents the *symplectic geometry*, while the right branch (or, right 1-cell TTM) represents the *Riemannian geometry*. Intuitively, the force functor $\mathcal{F}_1$ projects symplectic forces and torques into associated Riemannian accelerations (both linear and angular). rigorously, $\mathcal{F}_1$ maps the category of symplectic manifolds and their associated cotangent bundles into the category of Riemannian manifolds and their associated tangent bundles.

Inductively (see Appendix), a 2-functor $\mathcal{F}_2$ is a map from a *source 2-cell* to an *associated target 2-cell*.¹⁷

$$\begin{array}{ccc}
\begin{array}{ccc} & f & \\ \curvearrowright & & \curvearrowleft \\ A & \Downarrow \alpha & B \\ \curvearrowleft & & \curvearrowright \\ & g & \end{array} & \xrightarrow{\mathcal{F}_2} & \begin{array}{ccc} & \mathcal{F}_2(f) & \\ \curvearrowright & & \curvearrowleft \\ \mathcal{F}_2(A) & \Downarrow \mathcal{F}_2(\alpha) & \mathcal{F}_2(B) \\ \curvearrowleft & & \curvearrowright \\ & \mathcal{F}_2(g) & \end{array}
\end{array}$$

while a 3-functor $\mathcal{F}_3$ is a map from a source 3-cell to an associated target 3-cell:

$$\begin{array}{ccc}
\begin{array}{ccc} & f & \\ \curvearrowright & & \curvearrowleft \\ A & \alpha \begin{array}{c} \curvearrowright \psi \curvearrowleft \\ \rightleftarrows \end{array} \beta & B \\ \curvearrowleft & & \curvearrowright \\ & g & \end{array} & \xrightarrow{\mathcal{F}_3} & \begin{array}{ccc} & \mathcal{F}_3(f) & \\ \curvearrowright & & \curvearrowleft \\ \mathcal{F}_3(A) & \mathcal{F}_3(\alpha) \begin{array}{c} \curvearrowright \mathcal{F}_3(\psi) \curvearrowleft \\ \rightleftarrows \end{array} \mathcal{F}_3(\beta) & \mathcal{F}_3(B) \\ \curvearrowleft & & \curvearrowright \\ & \mathcal{F}_3(g) & \end{array}
\end{array}$$

etc.

¹⁶ Here, g_{ij} is the inertia matrix (or, material metric tensors including products and moments of inertia for all body segments), x^i are joint angles, L is the Legendre transformation, and $\mathcal{F}_1 : TT^*M \rightarrow TTM$ is the *covariant force functor*.

¹⁷ More properly, a 2-functor $\mathcal{F}_2$ is a map from a *source 2-category* represented by a l.h.s. 2-cell to a target 2-category represented by a r.h.s. 2-cell. Here, A, B are 1-cells, etc.

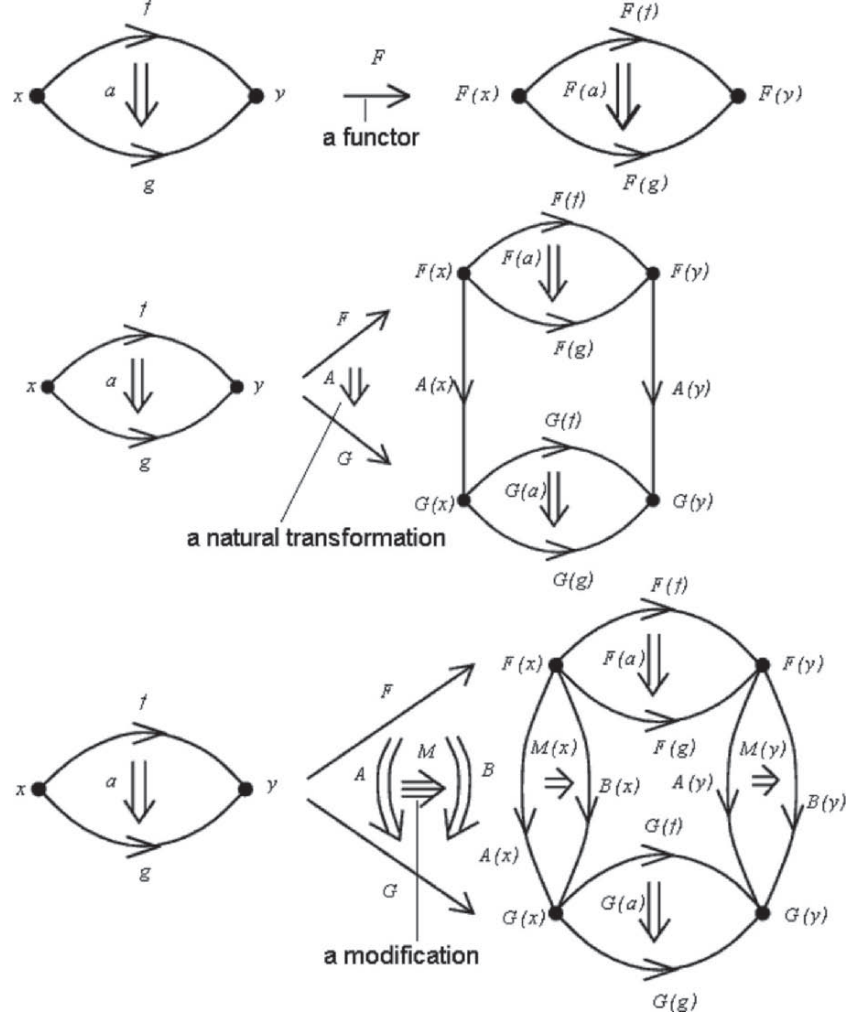


Fig. 1.23. Natural self-similarity of n -functors: a couple of 2-functors F, G together with natural transformations A, B and a modification M between them.

Modern mathematical sciences are abundant with n -functors: e.g., higher homotopy theory in modern topology,¹⁸ higher-order tangency (or contact) in differential geometry of jet bundles,¹⁹ (see [II06b] for technical details), etc.

¹⁸ In higher homotopy theory, we have some topological space X , in which 0-cells are *points* in X ; 1-cells are *paths* connecting endpoints in X (i.e., parameterized continuous maps $f : [0, 1] \rightarrow X$); 2-cells are *homotopies of paths* relative to endpoints (i.e., parameterized continuous maps $h : [0, 1] \times [0, 1] \rightarrow X$); 3-cells are *homotopies of homotopies* of paths in X (i.e., parameterized continuous maps $k : [0, 1] \times [0, 1] \times [0, 1] \rightarrow X$).

¹⁹ The concept of a *jet bundle* is based on the idea of *higher-order tangency*, or *higher-order contact*, at some designated point on a smooth manifold. Namely, a

All these n -functors are powerful modeling tools for both intuitive/conceptual and rigorous/predictive modeling of complex systems of virtually any nature.

Moreover, there are also rigorously defined *associations between any two n -functors of the same variance* (see Appendix), most notably *natural transformations* and *modifications*, as depicted in Figure 1.23 (for the case of 2-functors), showing their natural recursive self-similarity.

Also, recall from Appendix that the most important functorial operation is the *adjunction*, the ‘weak functorial inverse’ denoted $\mathcal{A} : \mathcal{F} \dashv \mathcal{G}$, representing the *association* between the two functors of the *opposite variance*, $(\mathcal{F}, \mathcal{G}) : \mathcal{S} \rightleftharpoons \mathcal{T}$ between the source category $\mathcal{S}$ and the target category $\mathcal{T}$.

Finally, the *composition of functors* is associative in the formal sense. For any three n -categories A, B, C we have the associative composition (chain) of n -functors F, G :

$$A \xrightarrow{F} B \xrightarrow{G} C = A \xrightarrow{F \circ G} C.$$

More generally, we have a chain:

$$A \xrightarrow{F} B \xrightarrow{G} C \xrightarrow{H} D = A \xrightarrow{F \circ G \circ H} D, \quad \text{etc.}$$

1.4.1 Preliminary Examples of Brain and Cognition Functors

Neurophysiological Functors

Firstly, an obvious neuro-physiological example of a *natural transformation* is an autonomic nervous system:

$$Aut.Nerv.Syst. : Sympathet \rightleftharpoons Parasympat,$$

which correlates the two functorial neural pathways

$$(Sympathet, Parasympat) : Brain \rightleftharpoons Body,$$

pair of smooth manifold maps from a source manifold M to a target manifold N ,

$$f_1, f_2 : M \rightarrow N,$$

are said to be k -*tangent* (or *tangent of order k* , or have a k th *order contact*) at a point x on a domain manifold M , denoted by $f_1 \sim f_2$, iff

$$\begin{aligned} f_1(x) &= f_2(x) && \text{called } 0\text{-tangent,} \\ \partial_x f_1(x) &= \partial_x f_2(x), && \text{called } 1\text{-tangent,} \\ \partial_{xx} f_1(x) &= \partial_{xx} f_2(x), && \text{called } 2\text{-tangent,} \\ \partial_{xxx} f_1(x) &= \partial_{xxx} f_2(x), && \text{called } 3\text{-tangent, etc.} \end{aligned}$$

from the source category *Brain* to the target category *Body*.

Secondly, recall from [II06a] that the core of the higher biodynamics is the global sensory–motor coordination/control, which in functorial language defines the *neuro–muscular adjunction*:

$$\textit{Coordination} : \textit{Sensory Path} \dashv \textit{Motor Path},$$

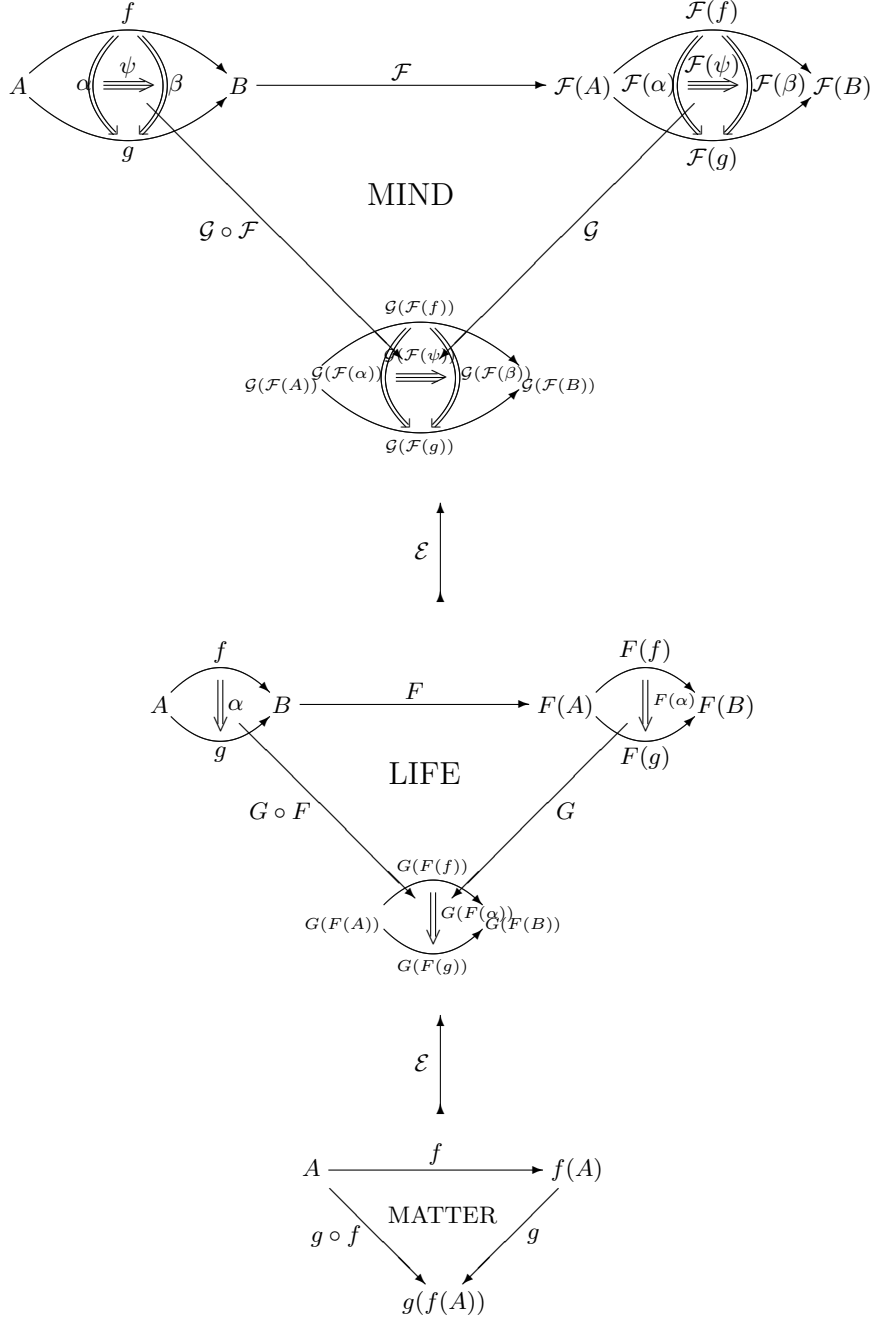
between the two functorial neural pathways

$$(\textit{Sensory}, \textit{Motor}) : \textit{Brain} \rightleftarrows \textit{Body},$$

between the same source (*Brain*) and target (*Body*) categories as before.

Emerging Functorial Structure $\mathcal{E} : \text{MATTER} \Rightarrow \text{LIFE} \Rightarrow \text{MIND}$

The 3–categorical MIND–structure, emerges from the 2–categorical LIFE–structure, which itself emerges from the 1–categorical MATTER–structure:



Neurodynamic Evolution 2-Functor $\mathcal{E}$

The goal-directed cognitive neurodynamics is (in subsection 3.5.7 below) defined as an evolution 2-functor $\mathcal{E}$, given by

$$\begin{array}{ccc}
 \begin{array}{ccc}
 A & \xrightarrow{f} & B \\
 \downarrow h & \text{CURRENT} & \downarrow g \\
 & \text{NEURAL} & \\
 & \text{STATE} & \\
 C & \xrightarrow{k} & D
 \end{array} & \xrightarrow{\mathcal{E}} & \begin{array}{ccc}
 \mathcal{E}(A) & \xrightarrow{\mathcal{E}(f)} & \mathcal{E}(B) \\
 \downarrow \mathcal{E}(h) & \text{DESIRED} & \downarrow \mathcal{E}(g) \\
 & \text{NEURAL} & \\
 & \text{STATE} & \\
 \mathcal{E}(C) & \xrightarrow{\mathcal{E}(k)} & \mathcal{E}(D)
 \end{array}
 \end{array}$$

Here $\mathcal{E}$ represents an association/projection functor from the *source 2-category* of the current neural state, defined as a commutative square of small categories $A, B, C, D, \dots$ of *current neural ensembles* and their causal interrelations $f, g, h, k, \dots$, onto the *target 2-category* of the desired neural state, defined as a commutative square of small categories $\mathcal{E}(A), \mathcal{E}(B), \mathcal{E}(C), \mathcal{E}(D), \dots$ of *evolved neural ensembles* and their causal interrelations $\mathcal{E}(f), \mathcal{E}(g), \mathcal{E}(h), \mathcal{E}(k)$. Each causal arrow in above diagram, e.g., $f : A \rightarrow B$, stands for a generic ‘neuro-morphism’, representing a self-organized, oscillatory neurodynamic system (see Chapter 3 for further details).

Various Approaches (Functors) to Brain and Cognition Modeling

Recall that *modeling* is the name of the game in any intelligence, be it human or machine, modeling for the purpose of *prediction* and *control* of a *complex system/process*, possibly within an *uncertain environment* (see [II06b]). With the good model and its exercising we can look forward in time with predictions and prescriptions and backward in time with diagnostics and explanations. With these *continuous-time* binding information structures we can make decisions and estimations in here and now for purposes of efficiency, efficacy and control into the future (see [Kos99]). In this Chapter we review various approaches (functors) to brain and cognition modeling. Material presented in this Chapter is the necessary background for comprehensive reading of the subsequent Chapters.

2.1 Nonlinear Dynamics and Chaos Approach

Above all, brain is a highly nonlinear dynamical system. In this section we give a brief review on *nonlinear dynamics and chaos*, which appears to be essential tool in modeling brain dynamics (see [Izh99b] and references therein). Formally, here we are dealing with a *dynamical evolution* 2-functor $\mathcal{E}$, given by

$$\begin{array}{ccc}
 \begin{array}{ccc}
 A & \xrightarrow{f} & B \\
 \downarrow h & \text{CURRENT} & \downarrow g \\
 & \text{DYNAMIC} & \\
 C & \xrightarrow{k} & D \\
 & \text{STATE} &
 \end{array}
 & \xrightarrow{\mathcal{E}} &
 \begin{array}{ccc}
 \mathcal{E}(A) & \xrightarrow{\mathcal{E}(f)} & \mathcal{E}(B) \\
 \downarrow \mathcal{E}(h) & \text{DESIRED} & \downarrow \mathcal{E}(g) \\
 & \text{DYNAMIC} & \\
 \mathcal{E}(C) & \xrightarrow{\mathcal{E}(k)} & \mathcal{E}(D) \\
 & \text{STATE} &
 \end{array}
 \end{array}$$

Here $\mathcal{E}$ represents an association/projection functor from the source 2-category of the current dynamical state, defined as a commutative square of small dynamical categories $A, B, C, D, \dots$ of *current dynamical components* and their

causal interrelations $f, g, h, k, \dots$, onto the *target 2-category* of the desired dynamical state, defined as a commutative square of small dynamical categories $\mathcal{E}(A), \mathcal{E}(B), \mathcal{E}(C), \mathcal{E}(D), \dots$ of *evolved dynamical components* and their causal interrelations $\mathcal{E}(f), \mathcal{E}(g), \mathcal{E}(h), \mathcal{E}(k)$. Each causal arrow in above diagram, e.g., $f : A \rightarrow B$, stands for a *generic dynamorphism*.¹

2.1.1 Chaos Theory Basics

Recall from [II06b] that the paradigm of *dynamical system* has its origins in *Newtonian mechanics*. There, as in other natural sciences and engineering disciplines, the evolution rule of dynamical systems is given implicitly by a relation that gives the state of the system only a short time into the future. This relation is either a differential equation or difference equation. To determine the state for all future times requires iterating the relation many times—each advancing time a small step. The iteration procedure is referred to as solving the system or integrating the system. Once the system can be solved, given an initial point it is possible to determine all its future points, a collection known as a *trajectory* or *orbit*. All possible system trajectories comprise its *flow* in the phase-space.

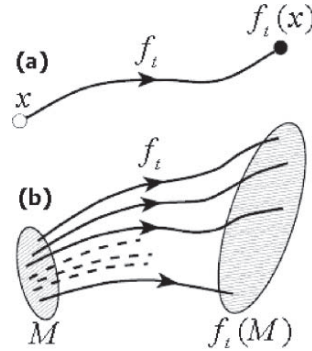


Fig. 2.1. Action of the *phase-flow* f_t in the phase-space manifold M : (a) Trajectory of a single initial point $x(t) \in M$, (b) Transporting the whole manifold M .

More precisely, recall from [II06b] that a *dynamical system* geometrically represents a *vector-field* (or, more generally, a *tensor-field*) in the system's phase-space manifold M , which upon *integration* (governed by the celebrated *existence and uniqueness theorems for ODEs*) defines a *phase-flow* in M (see Figure 2.1). This phase-flow $f_t \in M$, describing the complete behavior of a dynamical system at every time instant, can be either linear, nonlinear or chaotic.

¹ The concept of a *dynamorphism*, or a homomorphism between dynamical systems, has played an important role in theories of abstract automata [AAM76].

Before the advent of fast computers, solving a dynamical system required sophisticated mathematical techniques and could only be accomplished for a small class of linear dynamical systems. Numerical methods executed on computers have simplified the task of determining the orbits of a dynamical system.

For simple dynamical systems, knowing the trajectory is often sufficient, but most dynamical systems are too complicated to be understood in terms of individual trajectories. The difficulties arise because:

1. The systems studied may only be known approximately—the parameters of the system may not be known precisely or terms may be missing from the equations. The approximations used bring into question the validity or relevance of numerical solutions. To address these questions several notions of stability have been introduced in the study of dynamical systems, such as *Lyapunov stability* or *structural stability*. The stability of the dynamical system implies that there is a class of models or initial conditions for which the trajectories would be equivalent. The operation for comparing orbits to establish their equivalence changes with the different notions of stability.

2. The type of trajectory may be more important than one particular trajectory. Some trajectories may be periodic, whereas others may wander through many different states of the system. Applications often require enumerating these classes or maintaining the system within one class. Classifying all possible trajectories has led to the qualitative study of dynamical systems, that is, properties that do not change under coordinate changes. Linear dynamical systems and systems that have two numbers describing a state are examples of dynamical systems where the possible classes of orbits are understood.

3. The behavior of trajectories as a function of a parameter may be what is needed for an application. As a parameter is varied, the dynamical systems may have *bifurcation points* where the qualitative behavior of the dynamical system changes. For example, it may go from having only periodic motions to apparently erratic behavior, as in the transition to *turbulence* of a fluid.

4. The trajectories of the system may appear erratic, as if random. In these cases it may be necessary to compute averages using one very long trajectory or many different trajectories. The averages are well defined for ergodic systems and a more detailed understanding has been worked out for *hyperbolic systems*. Understanding the probabilistic aspects of dynamical systems has helped establish the foundations of statistical mechanics and of chaos.

Recall that a dynamical system may be defined as a deterministic rule for the time evolution of state observables. Well known examples are *ODEs* in which time is continuous,

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t)), \quad (\mathbf{x}, \mathbf{f} \in \mathbb{R}^n); \quad (2.1)$$

and *iterative maps* in which time is discrete:

$$\mathbf{x}(t+1) = \mathbf{g}(\mathbf{x}(t)), \quad (\mathbf{x}, \mathbf{g} \in \mathbb{R}^n). \quad (2.2)$$

In the case of maps, the evolution law is straightforward: from $\mathbf{x}(0)$ one computes $\mathbf{x}(1)$, and then $\mathbf{x}(2)$ and so on. For ODE's, under rather general assumptions on $\mathbf{f}$, from an initial condition $\mathbf{x}(0)$ one has a unique trajectory $\mathbf{x}(t)$ for $t > 0$ [Ott93]. Examples of regular behaviors (e.g., stable fixed-points, limit cycles, see Figure 2.2) are well known.

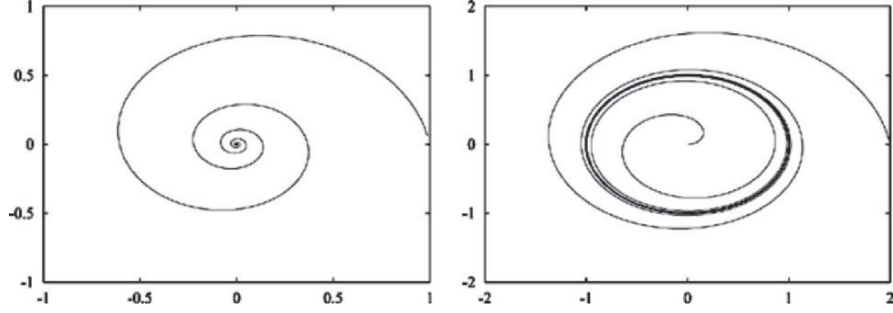


Fig. 2.2. Examples of regular attractors: fixed-point (left) and limit cycle (right). Note that limit cycles exist only in nonlinear dynamics.

A rather natural question is the possible existence of less regular behaviors, i.e., different from stable fixed-points, periodic or quasi-periodic motion. After the seminal works of Poincaré, Lorenz, Smale, May, and Hénon (to cite only the most eminent founders of *chaos theory*) it is now well established that the so called *chaotic behavior* is ubiquitous. As a relevant system, originated in the geophysical context, we mention the celebrated *Lorenz system* [Lor63, Spa82]

$$\begin{aligned}\dot{x} &= -\sigma(x - y) \\ \dot{y} &= -xz + rx - y \\ \dot{z} &= xy - bz\end{aligned}\tag{2.3}$$

This system is related to the *Rayleigh-Bénard convection* under very crude approximations (see Figure 2.3).² The quantity x is proportional to the circulatory fluid particle velocity; the quantities y and z are related to the temperature profile; σ , b and r are dimensionless parameters. Lorenz studied the case with $\sigma = 10$ and $b = 8/3$ at varying r (which is proportional to the Rayleigh number). It is easy to see by linear analysis that the fixed-point $(0, 0, 0)$ is stable for $r < 1$. For $r > 1$ it becomes unstable and two new fixed-points appear

$$C_{+,-} = (\pm\sqrt{b(r-1)}, \pm\sqrt{b(r-1)}, r-1),\tag{2.4}$$

² In 1963, Ed Lorenz from MIT was trying to improve weather forecasting. Using a primitive computer of those days, he discovered the first *chaotic attractor*. Lorenz used three Cartesian variables, (x, y, z) , to define *atmospheric convection*. Changing in time, these variables gave him a trajectory in a (Euclidean) 3D-space. From all starts, trajectories settle onto a chaotic, or *strange attractor*.

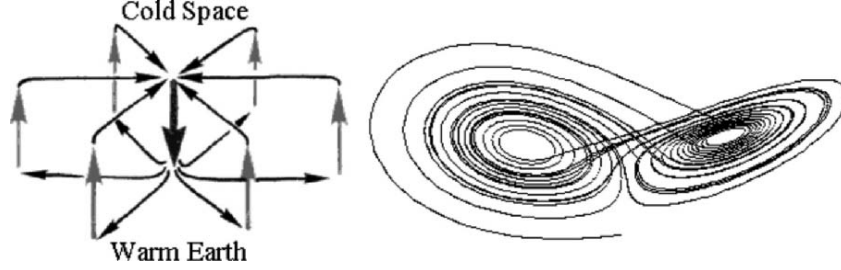


Fig. 2.3. Bénard cells, showing a typical vortex of a rolling air, with a warm air rising in a ring and a cool air descending in the center (left). A simple model of the Bénard cells provided by the celebrated ‘Lorenz–butterfly’ *strange attractor* (right).

these are stable for $r < r_c = 24.74$. A nontrivial behavior, i.e., non periodic, is present for $r > r_c$, as is shown in Figure 2.4.

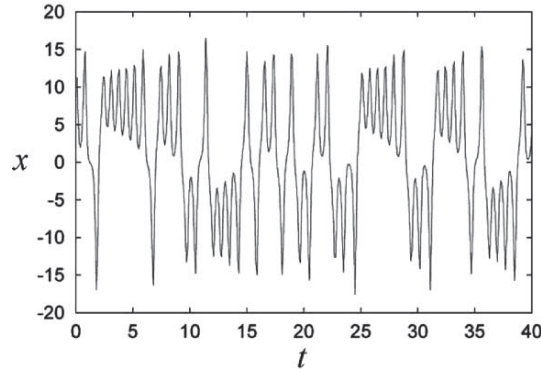


Fig. 2.4. Example of an aperiodic signal: the x variable of the Lorenz system (2.3) as function of time t , for $r = 28$.

In this ‘strange’, chaotic regime one has the so called *sensitive dependence on initial conditions*. Consider two trajectories, $\mathbf{x}(t)$ and $\mathbf{x}'(t)$, initially very close and denote with $\Delta(t) = \|\mathbf{x}'(t) - \mathbf{x}(t)\|$ their separation. Chaotic behavior means that if $\Delta(0) \rightarrow 0$, then as $t \rightarrow \infty$ one has $\Delta(t) \sim \Delta(0) \exp \lambda_1 t$, with $\lambda_1 > 0$ (see, e.g., [GOY87, YAS96, BG96, BLV01]).

This modern scientific term *deterministic chaos* depicts an *irregular and unpredictable* time evolution of many (simple) deterministic dynamical systems, characterized by nonlinear coupling of its variables (see, e.g., [GOY87, YAS96, BG96, Str94]). Given an initial condition, the dynamic equation determines the dynamic process, i.e., every step in the evolution. However, the initial condition, when magnified, reveals a cluster of values within a certain error bound. For a regular dynamic system, processes issuing from the cluster

are bundled together, and the bundle constitutes a predictable process with an error bound similar to that of the initial condition. In a chaotic dynamic system, processes issuing from the cluster diverge from each other exponentially, and after a while the error becomes so large that the dynamic equation loses its predictive power (see Figure 2.5).

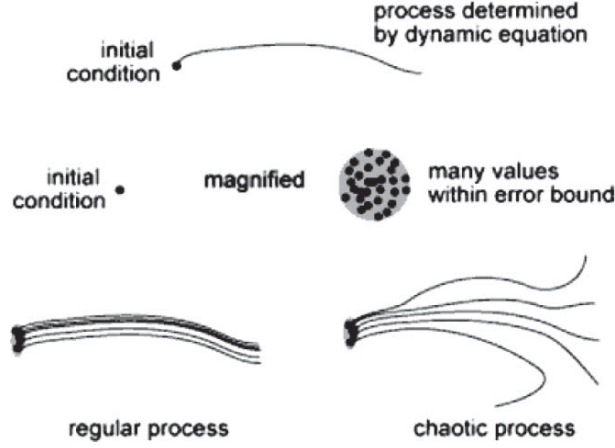


Fig. 2.5. Regular v.s. chaotic process.

For example, in a *pinball game*, any two trajectories that start out very close to each other separate exponentially with time, and in a finite (and in practice, a very small) number of bounces their separation $\delta x(t)$ attains the magnitude of L , the characteristic linear extent of the whole system. This property of sensitivity to initial conditions can be quantified as

$$|\delta x(t)| \approx e^{\lambda t} |\delta x(0)|,$$

where λ , the *mean rate of separation of trajectories* of the system, is called the *Lyapunov exponent*. For any finite accuracy $|\delta x(0)| = \delta x$ of the initial data, the dynamics is predictable only up to a finite *Lyapunov time*

$$T_{Lyap} \approx -\frac{1}{\lambda} \ln |\delta x/L|,$$

despite the deterministic and infallible simple laws that rule the pinball motion.

However, a positive Lyapunov exponent does not in itself lead to chaos (see [CAM05]). One could try to play 1- or 2-disk pinball game, but it would not be much of a game; trajectories would only separate, never to meet again. What is also needed is mixing, the coming together again and again of trajectories. While locally the nearby trajectories separate, the interesting dynamics

is confined to a globally finite region of the phase-space and thus the separated trajectories are necessarily folded back and can re-approach each other arbitrarily closely, infinitely many times. For the case at hand there are 2^n topologically distinct n bounce trajectories that originate from a given disk. More generally, the number of distinct trajectories with n bounces can be quantified as

$$N(n) \approx e^{hn},$$

where the *topological entropy* h ($h = \ln 2$ in the case at hand) is the growth rate of the number of topologically distinct trajectories.

When a physicist says that a certain system “exhibits chaos”, he means that the system obeys deterministic laws of evolution, but that the outcome is highly sensitive to small uncertainties in the specification of the initial state. The word “chaos” has in this context taken on a narrow technical meaning. If a deterministic system is locally unstable (positive Lyapunov exponent) and globally mixing (positive entropy), it is said to be *chaotic system*.

While mathematically correct, the definition of chaos as “positive Lyapunov exponent + positive entropy” is useless in practice, as a measurement of these quantities is intrinsically asymptotic and beyond reach for systems observed in nature. More powerful is Poincaré’s vision of chaos as the interplay of local instability (unstable periodic orbits) and global mixing (intertwining of their stable and unstable manifolds). In a chaotic system any open ball of initial conditions, no matter how small, will in finite time overlap with any other finite region and in this sense spread over the extent of the entire asymptotically accessible phase-space. Once this is grasped, the focus of theory shifts from attempting to predict individual trajectories (which is impossible) to a description of the geometry of the space of possible outcomes, and evaluation of averages over this space.

A definition of “turbulence” is even harder to come by. Intuitively, the word refers to *irregular behavior of an infinite-dimensional dynamical system described by deterministic equations of motion* – say, a bucket of boiling water – *described by the Navier–Stokes equations*. But in practice the word “turbulence” tends to refer to messy dynamics which we understand poorly. As soon as a phenomenon is understood better, it is reclaimed renamed as: “a route to chaos”, or “spatio-temporal chaos”, etc. (see [CAM05]).

Lyapunov Exponents

The sensitive dependence on the initial conditions can be formalized in order to give it a quantitative characterization. The main growth rate of trajectory separation is measured by the first (or maximum) *Lyapunov exponent*, defined as (see, e.g., [BLV01])

$$\lambda_1 = \lim_{t \rightarrow \infty} \lim_{\Delta(0) \rightarrow 0} \frac{1}{t} \ln \frac{\Delta(t)}{\Delta(0)}, \quad (2.5)$$

As long as $\Delta(t)$ remains sufficiently small (i.e., infinitesimal, strictly speaking), one can regard the separation as a tangent vector $\mathbf{z}(t)$ whose time evolution is

$$\dot{z}_i = \frac{\partial f_i}{\partial x_j} |_{\mathbf{x}(t)} \cdot z_j, \quad (2.6)$$

and, therefore,

$$\lambda_1 = \lim_{t \rightarrow \infty} \frac{1}{t} \ln \frac{\|\mathbf{z}(t)\|}{\|\mathbf{z}(0)\|}. \quad (2.7)$$

In principle, λ_1 may depend on the initial condition $\mathbf{x}(0)$, but this dependence disappears for ergodic systems. In general there exist as many Lyapunov exponents, conventionally written in decreasing order $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \dots$, as the independent coordinates of the phase-space [BGG80]. Without entering the details, one can define the sum of the first k Lyapunov exponents as the growth rate of an infinitesimal kD volume in the phase-space. In particular, λ_1 is the growth rate of material lines, $\lambda_1 + \lambda_2$ is the growth rate of $2D$ surfaces, and so on. A numerical widely used efficient method is due to Benettin et al. [BGG80].

It must be observed that, after a transient, the growth rate of any generic small perturbation (i.e., distance between two initially close trajectories) is measured by the first (maximum) Lyapunov exponent λ_1 , and $\lambda_1 > 0$ means chaos. In such a case, the state of the system is unpredictable on long times. Indeed, if we want to predict the state with a certain tolerance Δ then our forecast cannot be pushed over a certain time interval T_P , called *predictability time*, given by [BLV01]:

$$T_P \sim \frac{1}{\lambda_1} \ln \frac{\Delta}{\Delta(0)}. \quad (2.8)$$

The above relation shows that T_P is basically determined by $1/\lambda_1$, seen its weak dependence on the ratio $\Delta/\Delta(0)$. To be precise one must state that, for a series of reasons, relation (2.8) is too simple to be of actual relevance [BCF02].

Kolmogorov–Sinai Entropy

Deterministic chaotic systems, because of their irregular behavior, have many aspects in common with stochastic processes. The idea of using stochastic processes to mimic chaotic behavior, therefore, is rather natural [Chi79, Ben84]. One of the most relevant and successful approaches is symbolic dynamics [BS93]. For the sake of simplicity let us consider a discrete time dynamical system. One can introduce a partition $\mathcal{A}$ of the phase-space formed by N disjoint sets $A_1, \dots, A_N$. From any initial condition one has a trajectory

$$\mathbf{x}(0) \rightarrow \mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(n), \dots \quad (2.9)$$

dependently on the partition element visited, the trajectory (2.9), is associated to a symbolic sequence

$$\mathbf{x}(0) \rightarrow i_1, i_2, \dots, i_n, \dots \quad (2.10)$$

where i_n ($n = 1, 2, \dots, N$) means that $\mathbf{x}(n) \in A_{i_n}$ at the step n , for $n = 1, 2, \dots$. The coarse-grained properties of chaotic trajectories are therefore studied through the discrete time process (2.10).

An important characterization of symbolic dynamics is given by the *Kolmogorov–Sinai entropy* (KS), defined as follows. Let $C_n = (i_1, i_2, \dots, i_n)$ be a generic ‘word’ of size n and $P(C_n)$ its occurrence probability, the quantity [BLV01]

$$H_n = \sup_A \left[- \sum_{C_n} P(C_n) \ln P(C_n) \right], \quad (2.11)$$

is called *block entropy* of the n –sequences, and it is computed by taking the largest value over all possible partitions. In the limit of infinitely long sequences, the asymptotic entropy increment

$$h_{KS} = \lim_{n \rightarrow \infty} H_{n+1} - H_n, \quad (2.12)$$

is the Kolmogorov–Sinai entropy. The difference $H_{n+1} - H_n$ has the intuitive meaning of average information gain supplied by the $(n+1)$ –th symbol, provided that the previous n symbols are known. KS–entropy has an important connection with the positive Lyapunov exponents of the system [Ott93]:

$$h_{KS} = \sum_{\lambda_i > 0} \lambda_i. \quad (2.13)$$

In particular, for low–dimensional chaotic systems for which only one Lyapunov exponent is positive, one has $h_{KS} = \lambda_1$.

We observe that in (2.11) there is a technical difficulty, i.e., taking the sup over all the possible partitions. However, sometimes there exists a special partition, called *generating partition*, for which one finds that H_n coincides with its superior bound. Unfortunately the generating partition is often hard to find, even admitting that it exist. Nevertheless, given a certain partition, chosen by physical intuition, the statistical properties of the related symbol sequences can give information on the dynamical system beneath. For example, if the probability of observing a symbol (state) depends only by the knowledge of the immediately preceding symbol, the symbolic process becomes a *Markov chain* (see [II06b]) and all the statistical properties are determined by the transition matrix elements W_{ij} giving the probability of observing a transition $i \rightarrow j$ in one time step. If the memory of the system extends far beyond the time step between two consecutive symbols, and the occurrence probability of a symbol depends on k preceding steps, the process is called *Markov process* of order k and, in principle, a k rank tensor would be required to describe the dynamical system with good accuracy. It is possible to demonstrate that if $H_{n+1} - H_n = h_{KS}$ for $n \geq k + 1$, k is the (minimum) order of the required Markov process [Khi57]. It has to be pointed out, however, that to know the order of the suitable Markov process we need is of no practical utility if $k \gg 1$.

Pinball Game and Periodic Orbits

Confronted with a potentially chaotic dynamical system, we analyze it through a sequence of three distinct stages: (i) diagnose, (ii) count, (iii) measure. First we determine the intrinsic dimension of the system – the minimum number of coordinates necessary to capture its essential dynamics. If the system is very turbulent we are, at present, out of luck. We know only how to deal with the transitional regime between regular motions and chaotic dynamics in a few dimensions. That is still something; even an infinite-dimensional system such as a burning flame front can turn out to have a very few chaotic degrees of freedom. In this regime the chaotic dynamics is restricted to a space of low dimension, the number of relevant parameters is small, and we can proceed to step (ii); we count and classify all possible topologically distinct trajectories of the system into a hierarchy whose successive layers require increased precision and patience on the part of the observer. If successful, we can proceed with step (iii): investigate the weights of the different pieces of the system [CAM05].

With the game of pinball we are lucky: it is only a 2D system, free motion in a plane. The motion of a point particle is such that after a collision with one disk it either continues to another disk or it escapes. If we label the three disks by 1, 2 and 3, we can associate every trajectory with an itinerary, a sequence of labels indicating the order in which the disks are visited; for example, the two trajectories in figure 1.2 have itineraries 2313, 23132321 respectively. The itinerary is finite for a scattering trajectory, coming in from infinity and escaping after a finite number of collisions, infinite for a trapped trajectory, and infinitely repeating for a periodic orbit.³ Such labeling is the simplest example of *symbolic dynamics*. As the particle cannot collide two times in succession with the same disk, any two consecutive symbols must differ. This is an example of *pruning*, a rule that forbids certain subsequences of symbols. Deriving pruning rules is in general a difficult problem, but with the game of pinball we are lucky, as there are no further pruning rules.⁴

Suppose you wanted to play a good game of pinball, that is, get the pinball to bounce as many times as you possibly can – what would be a winning strategy? The simplest thing would be to try to aim the pinball so it bounces many times between a pair of disks – if you managed to shoot it so it starts out in the periodic orbit bouncing along the line connecting two disk centers, it would stay there forever. Your game would be just as good if you managed to get it to keep bouncing between the three disks forever, or place it on any periodic orbit. The only rub is that any such orbit is unstable, so you have to aim very accurately in order to stay close to it for a while. So it is pretty clear

³ The words *orbit* and *trajectory* here are synonymous.

⁴ The choice of symbols is in no sense unique. For example, as at each bounce we can either proceed to the next disk or return to the previous disk, the above 3-letter alphabet can be replaced by a binary $\{0,1\}$ alphabet. A clever choice of an alphabet will incorporate important features of the dynamics, such as its symmetries.

that if one is interested in playing well, unstable periodic orbits are important – they form the skeleton onto which all trajectories trapped for long times cling.

Now, recall that a trajectory is *periodic* if it returns to its starting position and momentum. It is custom to refer to the set of periodic points that belong to a given periodic orbit as a *cycle*.

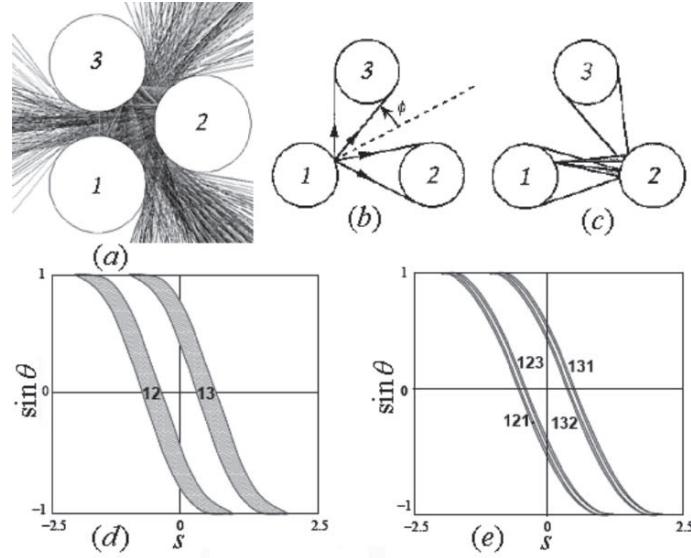


Fig. 2.6. A 3-disk pinball game. Up: (a) Elastic scattering around three hard disks (simulated in *Dynamics SolverTM*); (b) A trajectory starting out from disk 1 can either hit another disk or escape; (c) Hitting two disks in a sequence requires a much sharper aim; the cones of initial conditions that hit more and more consecutive disks are nested within each other. Down: Poincaré section for the 3-disk pinball game, with trajectories emanating from the disk 1 with $x_0 = (\text{arc-length}, \text{parallel momentum}) = (s_0, p_0)$, disk radius: center separation ratio $a : R = 1 : 2.5$; (d) Strips of initial points M_{12}, M_{13} which reach disks 2, 3 in one bounce, respectively. (e) Strips of initial points $M_{121}, M_{131}, M_{132}$ and M_{123} which reach disks 1, 2, 3 in two bounces, respectively; the Poincaré sections for trajectories originating on the other two disks are obtained by the appropriate relabeling of the strips (modified and adapted from [CAM05]).

Short periodic orbits are easily drawn and enumerated, but it is rather hard to perceive the systematics of orbits from their shapes. In mechanics a trajectory is fully and uniquely specified by its position and momentum at a given instant, and no two distinct phase-space trajectories can intersect. Their projections on arbitrary subspaces, however, can and do intersect, in rather unilluminating ways. In the pinball example, the problem is that we are looking at the projections of a 4D phase-space trajectories onto its

2D subspace, the configuration space. A clearer picture of the dynamics is obtained by constructing a phase-space Poincaré section.

The position of the ball is described by a pair of numbers (the spatial coordinates on the plane), and the angle of its velocity vector. As far as a classical dynamist is concerned, this is a complete description. Now, suppose that the pinball has just bounced off disk 1. Depending on its position and outgoing angle, it could proceed to either disk 2 or 3. Not much happens in between the bounces – the ball just travels at constant velocity along a straight line – so we can reduce the 4D flow to a 2D map f that takes the coordinates of the pinball from one disk edge to another disk edge. Let us state this more precisely: the trajectory just after the moment of impact is defined by marking s_n , the arc-length position of the n th bounce along the billiard wall, and $p_n = p \sin \phi_n$ is the momentum component parallel to the billiard wall at the point of impact (see Figure 2.6). Such a section of a flow is called a *Poincaré section*, and the particular choice of coordinates (due to Birkhoff) is particularly smart, as it conserves the phase-space volume. In terms of the Poincaré section, the dynamics is reduced to the *return map*

$$P : (s_n, p_n) \rightarrow (s_{n+1}, p_{n+1}),$$

from the boundary of a disk to the boundary of the next disk.

Next, we mark in the Poincaré section those initial conditions which do not escape in one bounce. There are two strips of survivors, as the trajectories originating from one disk can hit either of the other two disks, or escape without further ado. We label the two strips M_0, M_1 . Embedded within them there are four strips, $M_{00}, M_{10}, M_{01}, M_{11}$ of initial conditions that survive for two bounces, and so forth (see Figure 2.6). Provided that the disks are sufficiently separated, after n bounces the survivors are divided into 2^n distinct strips: the M_i th strip consists of all points with itinerary $i = s_1 s_2 s_3 \dots s_n$, $s = \{0, 1\}$. The unstable cycles as a skeleton of chaos are almost visible here: each such patch contains a periodic point $\overline{s_1 s_2 s_3 \dots s_n}$ with the basic block infinitely repeated. Periodic points are skeletal in the sense that as we look further and further, the strips shrink but the periodic points stay put forever.

We see now why it pays to utilize a symbolic dynamics; it provides a navigation chart through chaotic phase-space. There exists a unique trajectory for every admissible infinite length itinerary, and a unique itinerary labels every trapped trajectory. For example, the only trajectory labeled by 12 is the 2-cycle bouncing along the line connecting the centers of disks 1 and 2; any other trajectory starting out as 12 . . . either eventually escapes or hits the 3rd disk [CAM05].

Now we can ask what is a good physical quantity to compute for the game of pinball? Such system, for which almost any trajectory eventually leaves a finite region (the pinball table) never to return, is said to be open, or a *repeller*. The repeller escape rate is an eminently measurable quantity. An example of such a measurement would be an unstable molecular or nuclear state which

can be well approximated by a classical potential with the possibility of escape in certain directions. In an experiment many projectiles are injected into such a non-confining potential and their mean escape rate is measured. The numerical experiment might consist of injecting the pinball between the disks in some random direction and asking how many times the pinball bounces on the average before it escapes the region between the disks. On the other hand, for a theorist a good game of pinball consists in predicting accurately the asymptotic lifetime (or the escape rate) of the pinball.

Here we briefly show how Cvitanovic's *periodic orbit theory* [Cvi91] accomplishes this for us. Each step will be so simple that you can follow even at the cursory pace of this overview, and still the result is surprisingly elegant. Let us consider Figure 2.6 again. In each bounce, the initial conditions get thinned out, yielding twice as many thin strips as at the previous bounce. The total area that remains at a given time is the sum of the areas of the strips, so that the fraction of survivors after n bounces, or the *survival probability* is given by

$$\begin{aligned}\hat{\Gamma}_1 &= \frac{|M_0|}{|M|} + \frac{|M_1|}{|M|}, \\ \hat{\Gamma}_2 &= \frac{|M_{00}|}{|M|} + \frac{|M_{10}|}{|M|} + \frac{|M_{01}|}{|M|} + \frac{|M_{11}|}{|M|}, \\ &\dots \\ \hat{\Gamma}_n &= \frac{1}{|M|} \sum_{i=1}^{(n)} |M_i|,\end{aligned}\tag{2.14}$$

where $i = 01, 10, 11, \dots$ is a label of the i th strip (not a binary number), $|M|$ is the initial area, and $|M_i|$ is the area of the i th strip of survivors. Since at each bounce one routinely loses about the same fraction of trajectories, one expects the sum (2.14) to fall off exponentially with n and tend to the limit

$$\Gamma_{n+1}/\hat{\Gamma}_n = e^{-\gamma n} \rightarrow e^{-\gamma},$$

where the quantity γ is called the *escape rate* from the repeller. In [Cvi91] and subsequent papers, Cvitanovic has showed that the escape rate γ can be extracted from a highly convergent exact expansion by reformulating the sum (2.14) in terms of *unstable periodic orbits*.

Electronic Chaos: Chua's Circuit

Chua's circuit is a simple electronic circuit that exhibits classic chaotic behavior. First introduced in 1983 by Leon O. Chua, its ease of construction has made it an ubiquitous real-world example of a chaotic system, leading some to declare it 'a paradigm for chaos'. It has been the subject of much study; hundreds of papers have been published on this topic (see [Chu94]).

The *Chua's circuit* consists of two linear capacitors, two linear resistors, one linear inductor and a nonlinear resistor (see Figure 2.7). By varying the

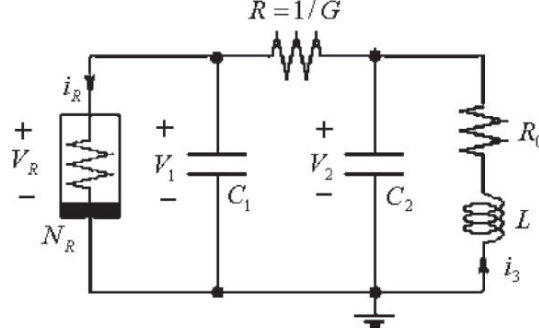


Fig. 2.7. Chua's circuit.

various circuit parameters, we can get complicated nonlinear and chaotic phenomena. Let us consider the case where we vary the conductance G of the resistor R and keep the other components fixed. In particular, we choose $L = 18\text{ mH}$, $R_0 = 12.5\text{ Ohms}$, $C_1 = 10\text{ nF}$, $C_2 = 100\text{ nF}$. The nonlinear resistor N_R (Chua's diode) is chosen to have a piecewise-linear $V-I$ characteristic of the form:

$$i = - \begin{cases} G_b v + G_a - G_b & \text{if } v > 1, \\ G_a v & \text{if } |v| < 1, \\ G_b v + G_b - G_a & \text{if } v < -1, \end{cases}$$

with $G_a = -0.75757\text{ mS}$, and $G_b = -0.40909\text{ mS}$.

Starting from low G -values, the circuit is stable and all trajectories converge towards one of the two stable equilibrium points. As G is increased, a limit cycle appears due to a *Hopf-like bifurcation*. In order to observe the period-doubling route to chaos, we need to further increase G . At the end of the period-doubling bifurcations, we observe a chaotic attractor. Because of symmetry, there exists a twin attractor lying in symmetrical position with respect the origin. As G is further increased, these two chaotic attractors collide and form a 'double scroll' chaotic attractor.

After normalization, the state equations for the Chua's circuit read:

$$\dot{x} = a(y - x - f(x)), \quad \dot{y} = x - y + z, \quad \dot{z} = -by - cz, \quad (2.15)$$

where $f(x)$ is a nonlinear function to be manipulated to give various chaotic behaviors.

By using a specific form of the nonlinearity $f(x)$, a family of *multi-spiral strange attractors* have been generated in [Ala99] (see Figure 2.8).

Phase Plane: Nonlinear Dynamics without Chaos

The general form of a 2D vector-field on the phase plane (similar to one in Figure 2.9) is given by

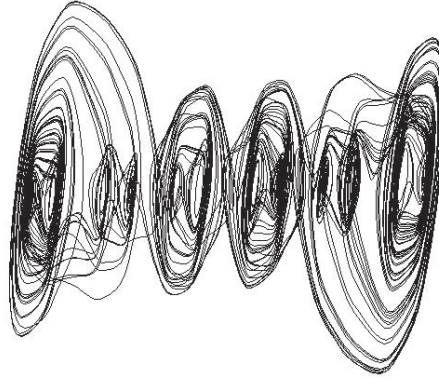


Fig. 2.8. A multi-spiral strange attractor of the Chua's circuit (modified from [Ala99]).

$$\dot{x}_1 = f_1(x_1, x_2), \quad \dot{x}_2 = f_2(x_1, x_2),$$

where f_i ($i = 1, 2$) are given function. By ‘flowing along’ the above vector-field, a *phase point* ‘traces out’ a solution $x_i(t)$, corresponding to a *trajectory* which is tangent to the vector-field. The entire phase plane is filled with trajectories (since each point can play the role of initial condition, depicting the so-called *phase portrait*. Every phase portrait has the following salient features (see [Str94]):

1. The fixed points, which satisfy: $f_i(x) = 0$, and correspond to the system's steady states or equilibria.
2. The closed orbits, corresponding to the *periodic solutions* (for which $x(t + T) = x(t)$, for all t , for some $T > 0$).
3. The specific *flow pattern*, i.e., the arrangement of trajectories near the fixed points and closed orbits.
4. The *stability* (attracting property) or *instability* (repelling property) of the fixed points and closed orbits.

Nothing more complicated than the fixed points and closed orbits can exist in the phase plane, according to the celebrated *Poincaré–Bendixson theorem*, which says that the dynamical possibilities in the phase plane are very limited. Specifically, *there cannot be chaotic behavior in the phase plane*. In other words, there is *no chaos in continuous 2D systems*.

However, there can exist chaotic behavior in *non-autonomous 2D continuous systems*, namely in the *forced nonlinear oscillators*, where explicit time-dependence actually represents the third dimension.

Free vs. Forced Nonlinear Oscillators

Here we give three examples of classical nonlinear oscillators, each in two modes: free (non-chaotic) and forced (possibly chaotic). For the simulation

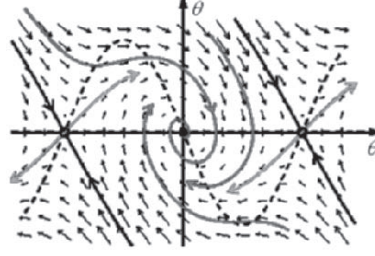


Fig. 2.9. Phase-portrait of a damped pendulum: Arrows denote the phase-flow, dashed line is a null-cline, filled dot is a stable fixed-point, open dot is an unstable fixed-point, dark gray curves are trajectories starting from sample initial points, dark lines with arrows are stable directions (manifolds), light lines with arrows are unstable directions (manifolds), the area between the stable manifolds is basin of attraction.

we use the technique called *time-phase plot*, combining an ordinary time plot with a phase-plane plot. We can see the considerable difference in complexity between unforced and forced oscillators (with all other parameters being the same). The reason for this is that *all forced 2D oscillators actually have dimension 3, although they are commonly written as a second-order ODE*. That is why for development of non-autonomous mechanics we use the *formalism of jet bundles*, see [II06b].

Spring

- Free (Rayleigh) spring (see Figure 2.10):

$$\begin{aligned}\dot{x} &= y, \\ \dot{y} &= -\frac{1}{m}(ax^3 + bx + cy),\end{aligned}$$

where x is displacement, y is velocity, $m > 0$ is mass, $ax^3 + bx + cy$ is the restoring force of the spring, with $b > 0$; we have three possible cases: hard spring ($a > 0$), linear (Hooke) spring ($a = 0$), or soft spring ($a < 0$).⁵

- Forced (Duffing) spring (see Figure 2.11):

$$\begin{aligned}\dot{x} &= y, \\ \dot{y} &= -\frac{1}{m}(ax^3 + bx + cy) + F \cos(wt), \\ \dot{\theta} &= w,\end{aligned}$$

where F is the force amplitude, θ is the driving phase and w is the driving frequency; the rest is the same as above.

⁵ In his book *The Theory of Sound*, Lord Rayleigh introduced a series of methods that would prove quite general, such as the notion of a *limit cycle* – a periodic motion a system goes to regardless of the initial conditions.

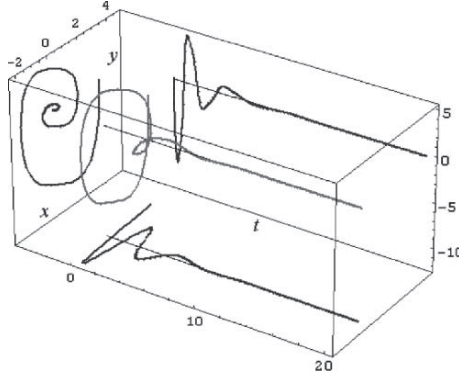


Fig. 2.10. Time-phase plot of the free hard spring with the following parameters: $m = 0.5 \text{ kg}$, $a = 1.3$, $b = 0.7$, $c = 0.5$, $x_0 = 3$, $y_0 = 0$, $t_{max} = 20 \text{ s}$. Simulated using *Mathematica*TM.

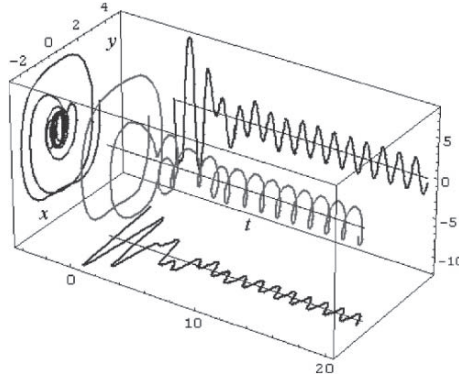


Fig. 2.11. Time-phase plot of the forced hard spring with the following parameters: $m = 0.5 \text{ kg}$, $a = 1.3$, $b = 0.7$, $c = 0.5$, $x_0 = 3$, $y_0 = 0$, $t_{max} = 20 \text{ s}$, $F = 10$, $w = 5$. Simulated using *Mathematica*TM.

Self-Sustained Oscillator

- Free (Rayleigh) self-sustained oscillator (see Figure 2.12):

$$\begin{aligned}\dot{x} &= y, \\ \dot{y} &= -\frac{1}{CL}(x + By^3 - Ay),\end{aligned}$$

where x is current, y is voltage, $C > 0$ is capacitance and $L > 0$ is inductance; $By^3 - Ay$ (with $A, B > 0$) is the characteristic function of vacuum tube.

- Forced (Rayleigh) self-sustained oscillator (see Figure 2.13):

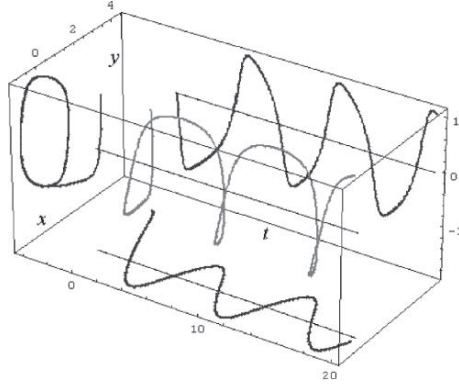


Fig. 2.12. Time-phase plot of the free Rayleigh's self-sustained oscillator with the following parameters: $A = 1.3$, $B = 1.5$, $C = 0.7$, $L = 1.5$, $x_0 = 3$, $y_0 = 0$, $t_{max} = 20$ s. Simulated using *Mathematica*TM.

$$\begin{aligned}\dot{x} &= y, \\ \dot{y} &= -\frac{1}{CL}(x + By^3 - Ay) + F \cos(wt), \\ \dot{\theta} &= w.\end{aligned}$$

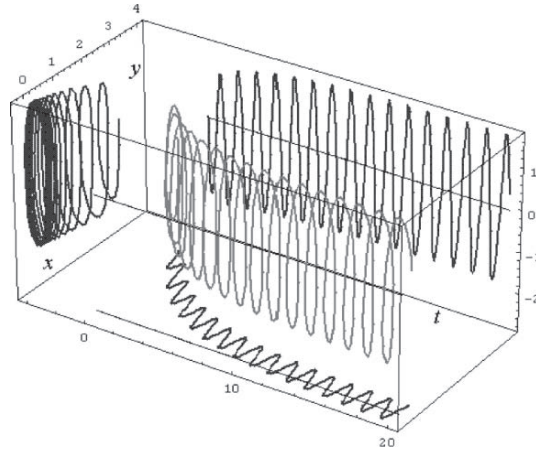


Fig. 2.13. Time-phase plot of the forced Rayleigh's self-sustained oscillator with the following parameters: $A = 1.3$, $B = 1.5$, $C = 0.7$, $L = 1.5$, $x_0 = 3$, $y_0 = 0$, $t_{max} = 20$ s, $F = 10$, $w = 5$. Simulated using *Mathematica*TM.

Van der Pol Oscillator

- Free Van der Pol oscillator (see Figure 2.14):

$$\begin{aligned}\dot{x} &= y, \\ \dot{y} &= -\frac{1}{CL}[x + (Bx^2 - A)y].\end{aligned}\tag{2.16}$$

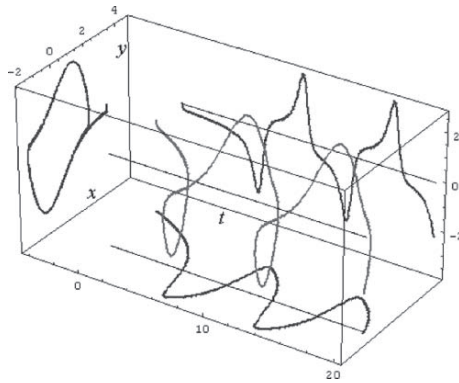


Fig. 2.14. Time-phase plot of the free Van der Pol oscillator with the following parameters: $A = 1.3$, $B = 1.5$, $C = 0.7$, $L = 1.5$, $x_0 = 3$, $y_0 = 0$, $t_{max} = 20$ s. Simulated using *Mathematica*TM.

- Forced Van der Pol oscillator (see Figure 2.15):

$$\begin{aligned}\dot{x} &= y, \\ \dot{y} &= -\frac{1}{CL}[x + (Bx^2 - A)y] + F \cos(wt), \\ \dot{\theta} &= w.\end{aligned}$$

Fractals: Mandelbrot and Julia Sets

Recall that *Mandelbrot and Julia sets* (see Figure 2.16) are celebrated *fractals*. Recall that fractals are sets with *fractional dimension* (see Figure 2.17). The Mandelbrot and Julia sets are defined either by a quadratic *conformal z -map* [Man80a, Man80b]

$$z_{n+1} = z_n^2 + c,$$

or by a real (x, y) -map

$$x_{n+1} = \sqrt{x_n} - \sqrt{y_n} + c_1, \quad y_{n+1} = 2x_n y_n + c_2,$$

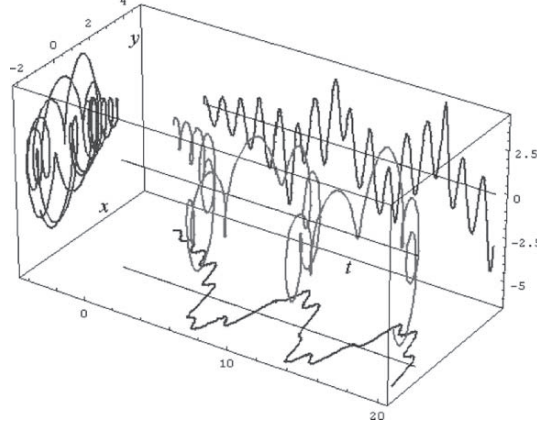


Fig. 2.15. Time-phase plot of the forced Van der Pol oscillator with the following parameters: $A = 1.3$, $B = 1.5$, $C = 0.7$, $L = 1.5$, $x_0 = 3$, $y_0 = 0$, $t_{max} = 20$ s, $F = 10$, $w = 5$. Simulated using *Mathematica*TM.

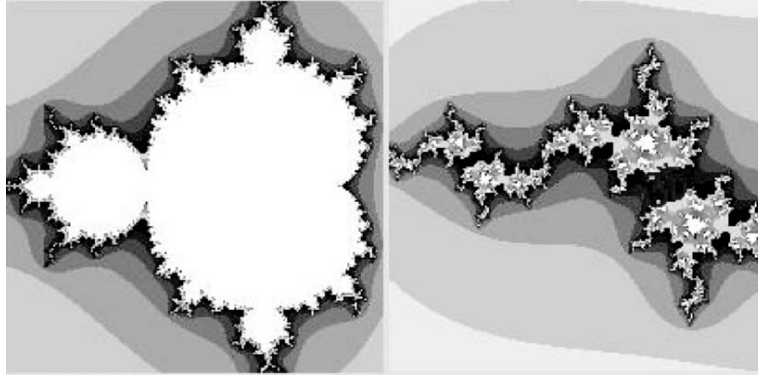


Fig. 2.16. The celebrated conformal Mandelbrot (left) and Julia (right) sets in the complex plane, simulated using *Dynamics Solver*TM.

where c, c_1 and c_2 are parameters. For almost every c , this conformal transformation generates a fractal (probably, only for $c = -2$ it is not a fractal). Julia set J_c with $c \ll 1$, the *capacity dimension* is

$$d_{cap} = 1 + \frac{|c|^2}{4 \ln 2} + O(|c|^3).$$

The set of all points for which J_c is connected is the Mandelbrot set.⁶

⁶ The Mandelbrot set has its place in complex-valued dynamics, a field first investigated by the French mathematicians Pierre Fatou and Gaston Julia at the beginning of the 20th century. For general families of holomorphic functions, the

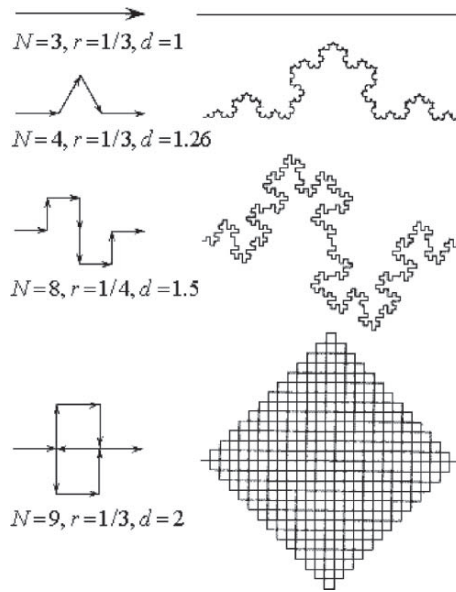


Fig. 2.17. Fractal dimension of curves in $\mathbb{R}^2$: $d = \frac{\log N}{\log 1/r}$.

2.1.2 Chaos Control

Feedback and Non-Feedback Algorithms for Chaos Control

Although the presence of chaotic behavior is generic and robust for suitable nonlinearities, ranges of parameters and external forces, there are practical situations where one wishes to avoid or control chaos so as to improve the performance of the dynamical system. Also, although chaos is sometimes useful as in a mixing process or in heat transfer, it is often unwanted or undesirable. For example, increased drag in flow systems, erratic fibrillations of heart beats, extreme weather patterns and complicated circuit oscillations are situations where chaos is harmful. Clearly, the ability to control chaos, that is to convert chaotic oscillations into desired regular ones with a periodic time dependence would be beneficial in working with a particular system. The possibility of purposeful selection and stabilization of particular orbits in a normally chaotic system, using minimal, predetermined efforts, provides a unique opportunity to maximize the output of a dynamical system. It is thus of great practical importance to develop suitable control methods and to analyze their efficacy.

boundary of the Mandelbrot set generalizes to the bifurcation locus, which is a natural object to study even when the connectedness locus is not useful. A related *Mandelbar set* was encountered by mathematician John Milnor in his study of parameter slices of real cubic polynomials; it is not locally connected; this property is inherited by the connectedness locus of real cubic polynomials.

Let us consider a general n D nonlinear dynamical system,

$$\dot{x} = F(x, p, t), \quad (2.17)$$

where $x = (x_1, x_2, x_3, \dots, x_n)$ represents the n state variables and p is a control or external parameter. Let $x(t)$ be a chaotic solution of (2.17). Different control algorithms are essentially based on the fact that one would like to effect the most minimal changes to the original system so that it will not be grossly deformed. From this point of view, controlling methods or algorithms can be broadly classified into two categories:

- (i) feedback methods, and
- (ii) non-feedback algorithms.

Feedback methods essentially make use of the intrinsic properties of chaotic systems, including their sensitivity to initial conditions, to stabilize orbits already existing in the systems. Some of the prominent methods are the following (see, [Lak97, Lak03, Sch88, II06b]):

1. Adaptive control algorithm;
2. Nonlinear control algorithm;
3. Ott–Grebogi–Yorke (OGY) method of stabilizing unstable periodic orbits;
4. Singer’s method of stabilizing unstable periodic orbits; and
5. Various control engineering approaches.

In contrast to feedback control techniques, non-feedback methods make use of a small perturbing external force such as a small driving force, a small noise term, a small constant bias or a weak modulation to some system parameter. These methods modify the underlying chaotic dynamical system weakly so that stable solutions appear. Some of the important controlling methods of this type are the following.

1. Parametric perturbation method
2. Addition of a weak periodic signal, constant bias or noise
3. Entrainment–open loop control
4. Oscillator absorber method.

Here is a typical example of adaptive control algorithm. We can control the chaotic orbit $X_s = (x_s, y_s)$ of the *Van der Pol oscillator* (see (2.71) below) by introducing the following dynamics on the parameter A_1 :

$$\begin{aligned} \dot{x} &= x - \frac{x^3}{3} - y + A_0 + A_1 \cos \omega t, & \dot{y} &= c(x + a - by), \\ \dot{A}_1 &= -\epsilon[(x - x_s) - (y - y_s)], & \epsilon &<< 1. \end{aligned}$$

On the other hand, recall from [II06b] that a generic SISO nonlinear system

$$\dot{x} = f(x) + g(x)u \quad y = h(x) \quad (2.18)$$

is said to have *relative degree* r at a point x^o if

- (i) $L_g L_f^k h(x) = 0$ for all x in a neighborhood of x^o and all $k < r - 1$
- (ii) $L_g L_f^{r-1} h(x^o) \neq 0$, where L_g denotes the *Lie derivative* in the direction of the vector-field g .

For example, the Van der Pol oscillator has the state space form

$$\dot{x} = f(x) + g(x)u = \begin{bmatrix} 2\omega\zeta(1 - \mu x_1^2)x_2 - \omega^2 x_1 \\ 1 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u. \quad (2.19)$$

Suppose the output function is chosen as

$$y = h(x) = x_1. \quad (2.20)$$

In this case we have

$$L_g h(x) = \frac{\partial h}{\partial x} g(x) = [1 \ 0] \begin{bmatrix} 0 \\ 1 \end{bmatrix} = 0, \quad \text{and} \quad (2.21)$$

$$L_f h(x) = \frac{\partial h}{\partial x} f(x) = [1 \ 0] \begin{bmatrix} 2\omega\zeta(1 - \mu x_1^2)x_2 - \omega^2 x_1 \\ 1 \end{bmatrix} = x_2. \quad (2.22)$$

Moreover

$$L_g L_f h(x) = \frac{\partial(L_f h)}{\partial x} g(x) = [0 \ 1] \begin{bmatrix} 0 \\ 1 \end{bmatrix} = 1 \quad (2.23)$$

and thus we see that the Van der Pol oscillator system has relative degree 2 at any point x^o .

However, if the output function is, for instance

$$y = h(x) = \sin x_2 \quad (2.24)$$

then $L_g h(x) = \cos x_2$. The system has relative degree 1 at any point x^o , provided that $(x^o)_2 \neq (2k+1)\pi/2$. If the point x^o is such that this condition is violated, no relative degree can be defined.

Both adaptive and nonlinear control methods can be naturally extended to other chaotic systems, e.g., *Lorenz attractor* (see Figure 2.18).

Hybrid Systems and Homotopy ODEs

Consider a *hybrid dynamical system of variable structure*, given by an n D ODE-system (see [MWH01])

$$\dot{x} = f(t, x), \quad (2.25)$$

where $x = x(t) \in \mathbb{R}^n$ and $f = f(t, x) : \mathbb{R}^+ \times \mathbb{R}^n \rightarrow \mathbb{R}^n$. Let the domain $G \subset \mathbb{R}^+ \times \mathbb{R}^n$, on which the vector-field $f(t, x)$ is defined, be divided into two subdomains, G^+ and G^- , by means of a smooth $(n-1)$ -manifold M . In $G^+ \cup M$, let there be given a vector-field $f^+(t, x)$, and in $G^- \cup M$, let there be given a vector-field $f^-(t, x)$. Assume that both $f^+ = f^+(t, x)$ and $f^- = f^-(t, x)$ are continuous in t and smooth in x . For the system (2.25), let

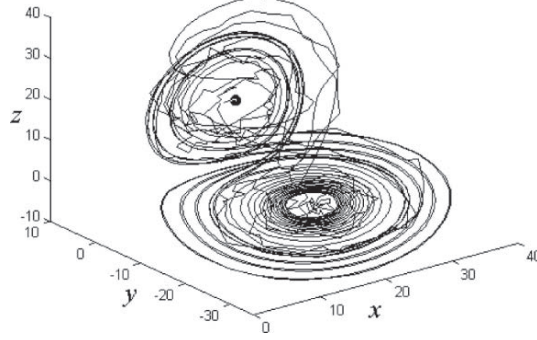


Fig. 2.18. Nonlinear control of the Lorenz system: targeting of unstable upper and lower states in the Lorenz attractor (after applying random perturbations, see [Pet96]), using a MIMO nonlinear controller (see [II06b]); simulated using *Matlab*TM.

$$f = \begin{cases} f^+ & \text{when } x \in G^+ \\ f^- & \text{when } x \in G^- \end{cases}.$$

Under these conditions, a solution $x(t)$ of ODE (2.25) is well-defined while passing through G until the manifold M is reached.

Upon reaching the manifold M , in physical systems with inertia, the transition

$$\text{from } \dot{x} = f^-(t, x) \quad \text{to} \quad \dot{x} = f^+(t, x)$$

does not take place instantly on reaching M , but after some delay. Due to this delay, the solution $x(t)$ oscillates about M , $x(t)$ being displaced along M with some mean velocity.

As the delay tends to zero, the limiting motion and velocity along M are determined by the *linear homotopy ODE*

$$\dot{x} = f^0(t, x) \equiv (1 - \alpha) f^-(t, x) + \alpha f^+(t, x), \quad (2.26)$$

where $x \in M$ and $\alpha \in [0, 1]$ is such that the *linear homotopy segment* $f^0(t, x)$ is tangential to M at the point x , i.e., $f^0(t, x) \in T_x M$, where $T_x M$ is the tangent space to the manifold M at the point x .

The vector-field $f^0(t, x)$ of the system (2.26) can be constructed as follows: at the point $x \in M$, $f^-(t, x)$ and $f^+(t, x)$ are given and their ends are joined by the linear homotopy segment. The point of intersection between this segment and $T_x M$ is the end of the required vector-field $f^0(t, x)$. The vector function $x(t)$ which satisfies (2.25) in G^- and G^+ , and (2.26) when $x \in M$, can be considered as a *solution* of (2.25) in a *general sense*.

However, there are cases in which the solution $x(t)$ cannot consist of a finite or even countable number of arcs, each of which passes through G^- or G^+ satisfying (2.25), or moves along the manifold M and satisfies the homotopic ODE (2.26). To cover such cases, assume that the vector-field $f = f(t, x)$ in

ODE (2.25) is a Lebesgue-measurable function in a domain $G \subset \mathbb{R}^+ \times \mathbb{R}^n$, and that for any closed bounded domain $D \subset G$ there exists a summable function $K(t)$ such that almost everywhere in D we have $|f(t, x)| \leq K(t)$. Then the absolutely continuous vector function $x(t)$ is called the *generalized solution* of the ODE (2.25) *in the sense of Filippov* (see [MWH01]) if for almost all t , the vector $\dot{x} = \dot{x}(t)$ belongs to the least convex closed set containing all the limiting values of the vector-field $f(t, x^*)$, where x^* tends towards x in an arbitrary manner, and the values of the function $f(t, x^*)$ on a set of measure zero in $\mathbb{R}^n$ are ignored.

Such *hybrid systems* of variable structure occur in the study of nonlinear electric networks (endowed with electronic switches, relays, diodes, rectifiers, etc.), in models of both natural and artificial neural networks, as well as in feedback control systems (usually with continuous-time plants and digital controllers/filters).

Exploiting Critical Sensitivity

The fact that some dynamical systems showing the necessary conditions for chaotic behavior possess such a critical dependence on the initial conditions was known since the end of the last century. However, only in the last thirty years, experimental observations have pointed out that, in fact, chaotic systems are common in nature. They can be found, e.g., in chemistry (*Belousov–Zhabotinski reaction*), in nonlinear optics (lasers), in electronics (*Chua–Matsumoto circuit*), in fluid dynamics (*Rayleigh–Bénard convection*), etc. Many natural phenomena can also be characterized as being chaotic. They can be found in meteorology, solar system, heart and brain of living organisms and so on.

Due to their critical dependence on the initial conditions, and due to the fact that, in general, experimental initial conditions are never known perfectly, these systems are intrinsically unpredictable. Indeed, the prediction trajectory emerging from an initial condition and the real trajectory emerging from the real initial condition diverge exponentially in course of time, so that the error in the prediction (the distance between prediction and real trajectories) grows exponentially in time, until making the system's real trajectory completely different from the predicted one at long times.

For many years, this feature made chaos undesirable, and most experimentalists considered such characteristic as something to be strongly avoided. Besides their critical sensitivity to initial conditions, chaotic systems exhibit two other important properties. Firstly, there is an infinite number of unstable periodic orbits embedded in the underlying chaotic set. In other words, the skeleton of a chaotic attractor is a collection of an infinite number of periodic orbits, each one being unstable. Secondly, the dynamics in the chaotic attractor is *ergodic*, which implies that during its temporal evolution the system ergodically visits small neighborhood of every point in each one of the unstable periodic orbits embedded within the chaotic attractor.

A relevant consequence of these properties is that a chaotic dynamics can be seen as shadowing some periodic behavior at a given time, and erratically jumping from one to another periodic orbit. The idea of controlling chaos is then when a trajectory approaches ergodically a desired periodic orbit embedded in the attractor, one applies small perturbations to stabilize such an orbit. If one switches on the stabilizing perturbations, the trajectory moves to the neighborhood of the desired periodic orbit that can now be stabilized. This fact has suggested the idea that the critical sensitivity of a chaotic system to changes (perturbations) in its initial conditions may be, in fact, very desirable in practical experimental situations. Indeed, if it is true that a small perturbation can give rise to a very large response in the course of time, it is also true that a judicious choice of such a perturbation can direct the trajectory to wherever one wants in the attractor, and to produce a series of desired dynamical states. This is exactly the idea of targeting [BGL00].

The important point here is that, because of chaos, one is able to produce an infinite number of desired dynamical behaviors (either periodic and not periodic) using the same chaotic system, with the only help of tiny perturbations chosen properly. We stress that this is not the case for a non-chaotic dynamics, wherein the perturbations to be done for producing a desired behavior must, in general, be of the same order of magnitude as the unperturbed evolution of the dynamical variables.

The *idea of chaos control* was enunciated in 1990 at the University of Maryland, by E. Ott, C. Grebogi and J.A. Yorke [OGY90], widely referred to as Ott–Grebogi–Yorke (OGY, for short). In OGY-paper [OGY90], the ideas for controlling chaos were outlined and a method for stabilizing an unstable periodic orbit was suggested, as a proof of principle. The main idea consisted in waiting for a natural passage of the chaotic orbit close to the desired periodic behavior, and then applying a small judiciously chosen perturbation, in order to stabilize such periodic dynamics (which would be, in fact, unstable for the unperturbed system). Through this mechanism, one can use a given laboratory system for producing an infinite number of different periodic behavior (the infinite number of its unstable periodic orbits), with a great flexibility in switching from one to another behavior. Much more, by constructing appropriate goal dynamics, compatible with the chaotic attractor, an operator may apply small perturbations to produce any kind of desired dynamics, even not periodic, with practical application in the coding process of signals.

A branch of the theory of dynamical systems has been developed with the aim of formalizing and quantitatively characterizing the sensitivity to initial conditions. The *largest Lyapunov exponent* λ (together with the related *Kaplan–Yorke dimension* d_{KY}) and the *Kolmogorov–Sinai entropy* h_{KS} are the two indicators for measuring the *rate of error growth* and *information* produced by the dynamical system [ER85].

Lyapunov Exponents and Kaplan–Yorke Dimension

The characteristic Lyapunov exponents are somehow an extension of the linear stability analysis to the case of aperiodic motions. Roughly speaking, they measure the typical rate of exponential divergence of nearby trajectories. In this sense they give information on the rate of growth of a very small error on the initial state of a system [BCF02].

Consider an n D dynamical system given by the set of ODEs of the form

$$\dot{x} = f(x), \quad (2.27)$$

where $x = (x_1, \dots, x_n) \in \mathbb{R}^n$ and $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$. Recall that since the r.h.s of equation (2.27) does not depend on t explicitly, the system is called *autonomous*. We assume that f is smooth enough that the evolution is well-defined for time intervals of arbitrary extension, and that the motion occurs in a bounded region R of the system phase-space M . We intend to study the separation between two trajectories in M , $x(t)$ and $x'(t)$, starting from two close initial conditions, $x(0)$ and $x'(0) = x(0) + \delta x(0)$ in $R_0 \subset M$, respectively.

As long as the difference between the trajectories, $\delta x(t) = x'(t) - x(t)$, remains infinitesimal, it can be regarded as a vector, $z(t)$, in the tangent space $T_x M$ of M . The time evolution of $z(t)$ is given by the linearized differential equations:

$$\dot{z}_i(t) = \left. \frac{\partial f_i}{\partial x_j} \right|_{x(t)} z_j(t).$$

Under rather general hypothesis, Oseledets [Ose68] proved that for almost all initial conditions $x(0) \in R$, there exists an orthonormal basis $\{e_i\}$ in the tangent space $T_x M$ such that, for large times,

$$z(t) = c_i e_i \exp(\lambda_i t), \quad (2.28)$$

where the coefficients $\{c_i\}$ depend on $z(0)$. The exponents $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d$ are called *characteristic Lyapunov exponents*. If the dynamical system has an ergodic invariant measure on M , the spectrum of LEs $\{\lambda_i\}$ does not depend on the initial conditions, except for a set of measure zero with respect to the natural invariant measure.

Equation (2.28) describes how a d D spherical region $R = S^n \subset M$, with radius ϵ centered in $x(0)$, deforms, with time, into an ellipsoid of semi-axes $\epsilon_i(t) = \epsilon \exp(\lambda_i t)$, directed along the e_i vectors. Furthermore, for a generic small perturbation $\delta x(0)$, the distance between the reference and the perturbed trajectory behaves as

$$|\delta x(t)| \sim |\delta x(0)| \exp(\lambda_1 t) [1 + O(\exp -(\lambda_1 - \lambda_2)t)].$$

If $\lambda_1 > 0$ we have a rapid (exponential) amplification of an error on the initial condition. In such a case, the system is chaotic and, unpredictable on the long times. Indeed, if the initial error amounts to $\delta_0 = |\delta x(0)|$, and we purpose to

predict the states of the system with a certain tolerance Δ , then the prediction is reliable just up to a *predictability time* given by

$$T_p \sim \frac{1}{\lambda_1} \ln \left(\frac{\Delta}{\delta_0} \right).$$

This equation shows that T_p is basically determined by the *positive leading Lyapunov exponent*, since its dependence on δ_0 and Δ is logarithmically weak. Because of its preeminent role, λ_1 is often referred as ‘the leading positive Lyapunov exponent’, and denoted by λ .

Therefore, Lyapunov exponents are average rates of expansion or contraction along the principal axes. For the i th principal axis, the corresponding Lyapunov exponent is defined as

$$\lambda_i = \lim_{t \rightarrow \infty} \{(1/t) \ln [L_i(t)/L_i(0)]\}, \quad (2.29)$$

where $L_i(t)$ is the radius of the ellipsoid along the i th principal axis at time t . For technical details on calculating Lyapunov exponents from any time series data, see [WSS85].

An initial volume V_0 of the phase-space region R_0 evolves on average as

$$V(t) = V_0 e^{(\lambda_1 + \lambda_2 + \dots + \lambda_{2n})t}, \quad (2.30)$$

and therefore the rate of change of $V(t)$ is simply

$$\dot{V}(t) = \sum_{i=1}^{2n} \lambda_i V(t).$$

In the case of a 2D phase area A , evolving as $A(t) = A_0 e^{(\lambda_1 + \lambda_2)t}$, a *Lyapunov dimension* d_L is defined as

$$d_L = \lim_{\epsilon \rightarrow 0} \left[\frac{d(\ln(N(\epsilon)))}{d(\ln(1/\epsilon))} \right],$$

where $N(\epsilon)$ is the number of squares with sides of length ϵ required to cover $A(t)$, and d represents an ordinary *capacity dimension*,

$$d_c = \lim_{\epsilon \rightarrow 0} \left(\frac{\ln N}{\ln(1/\epsilon)} \right).$$

Lyapunov dimension can be extended to the case of n D phase-space by means of the *Kaplan–Yorke dimension* [Kap00, YAS96, OGY90]) as

$$d_{KY} = j + \frac{\lambda_1 + \lambda_2 + \dots + \lambda_j}{|\lambda_{j+1}|},$$

where the λ_i are ordered (λ_1 being the largest) and j is the index of the smallest nonnegative Lyapunov exponent.

Kolmogorov–Sinai Entropy

The LE, λ , gives a first quantitative information on how rapidly we loose the ability of predicting the evolution of a system [BCF02]. A state, initially determined with an error $\delta x(0)$, after a time enough larger than $1/\lambda$, may be found almost everywhere in the region of motion $R \in M$. In this respect, the *Kolmogorov–Sinai* (KS) *entropy*, h_{KS} , supplies a more refined information. The error on the initial state is due to the maximal resolution we use for observing the system. For simplicity, let us assume the same resolution ϵ for each degree of freedom. We build a partition of the phase-space M with cells of volume ϵ^d , so that the state of the system at $t = t_0$ is found in a region R_0 of volume $V_0 = \epsilon^d$ around $x(t_0)$. Now we consider the trajectories starting from V_0 at t_0 and sampled at discrete times $t_j = j \tau$ ($j = 1, 2, 3, \dots, t$). Since we are considering motions that evolve in a bounded region $R \subset M$, all the trajectories visit a finite number of different cells, each one identified by a symbol. In this way a unique sequence of symbols $\{s(0), s(1), s(2), \dots\}$ is associated with a given trajectory $x(t)$. In a chaotic system, although each evolution $x(t)$ is univocally determined by $x(t_0)$, a great number of different symbolic sequences originates by the same initial cell, because of the divergence of nearby trajectories. The total number of the admissible symbolic sequences, $\tilde{N}(\epsilon, t)$, increases exponentially with a rate given by the topological entropy

$$h_T = \lim_{\epsilon \rightarrow 0} \lim_{t \rightarrow \infty} \frac{1}{t} \ln \tilde{N}(\epsilon, t).$$

However, if we consider only the number of sequences $N_{eff}(\epsilon, t) \leq \tilde{N}(\epsilon, t)$ which appear with very high probability in the long time limit – those that can be numerically or experimentally detected and that are associated with the natural measure – we arrive at a more physical quantity, namely the *Kolmogorov–Sinai entropy* [ER85]:

$$h_{KS} = \lim_{\epsilon \rightarrow 0} \lim_{t \rightarrow \infty} \frac{1}{t} \ln N_{eff}(\epsilon, t) \leq h_T. \quad (2.31)$$

h_{KS} quantifies the long time exponential rate of growth of the number of the effective coarse-grained trajectories of a system. This suggests a link with information theory where the Shannon entropy measures the mean asymptotic growth of the number of the typical sequences – the ensemble of which has probability almost one – emitted by a source.

We may wonder what is the number of cells where, at a time $t > t_0$, the points that evolved from R_0 can be found, i.e., we wish to know how big is the coarse-grained volume $V(\epsilon, t)$, occupied by the states evolved from the volume V_0 of the region R_0 , if the minimum volume we can observe is $V_{min} = \epsilon^d$. As stated above (2.30), we have

$$V(t) \sim V_0 \exp\left(t \sum_{i=1}^d \lambda_i\right).$$

However, this is true only in the limit $\epsilon \rightarrow 0$. In this (unrealistic) limit, $V(t) = V_0$ for a conservative system (where $\sum_{i=1}^d \lambda_i = 0$) and $V(t) < V_0$ for a dissipative system (where $\sum_{i=1}^d \lambda_i < 0$). As a consequence of limited resolution power, in the evolution of the volume $V_0 = \epsilon^d$ the effect of the contracting directions (associated with the negative Lyapunov exponents) is completely lost. We can experience only the effect of the expanding directions, associated with the positive Lyapunov exponents. As a consequence, in the typical case, the coarse grained volume behaves as

$$V(\epsilon, t) \sim V_0 e^{(\sum_{\lambda_i > 0} \lambda_i) t},$$

when V_0 is small enough. Since $N_{eff}(\epsilon, t) \propto V(\epsilon, t)/V_0$, one has

$$h_{KS} = \sum_{\lambda_i > 0} \lambda_i.$$

This argument can be made more rigorous with a proper mathematical definition of the metric entropy. In this case one derives the Pesin relation [Pes77, ER85]

$$h_{KS} \leq \sum_{\lambda_i > 0} \lambda_i. \quad (2.32)$$

Because of its relation with the Lyapunov exponents – or by the definition (2.31) – it is clear that also h_{KS} is a fine-grained and global characterization of a dynamical system.

The metric entropy is an invariant characteristic quantity of a dynamical system, i.e., given two systems with invariant measures, their KS-entropies exist and they are equal iff the systems are isomorphic [Bil65].

Chaos Control by Ott, Grebogi and Yorke (OGY)

Besides the occurrence of chaos in a large variety of natural processes, chaos may also occur because one may wish to design a physical, biological or chemical experiment, or to project an industrial plant to behave in a chaotic manner. The OGY-idea is that chaos may indeed be desirable since it can be controlled by using small perturbation to some accessible parameter.

The major key ingredient for the OGY-control of chaos is the observation that a chaotic set, on which the trajectory of the chaotic process lives, has embedded within it a large number of unstable low-period periodic orbits. In addition, because of ergodicity, the trajectory visits or accesses the neighborhood of each one of these periodic orbits. Some of these periodic orbits may correspond to a desired system's performance according to some criterion. The second ingredient is the realization that chaos, while signifying sensitive dependence on small changes to the current state and henceforth rendering unpredictable the system state in the long time, also implies that the system's behavior can be altered by using small perturbations. Then, the accessibility

of the chaotic systems to many different periodic orbits combined with its sensitivity to small perturbations allows for the control and the manipulation of the chaotic process. Specifically, the OGY approach is then as follows. One first determines some of the unstable low-period periodic orbits that are embedded in the chaotic set. One then examines the location and the stability of these orbits and chooses one which yields the desired system performance. Finally, one applies small control to stabilize this desired periodic orbit. However, all this can be done from data by using nonlinear time series analysis for the observation, understanding and control of the system. This is particularly important since chaotic systems are rather complicated and the detailed knowledge of the equations of the process is often unknown [BGL00].

Simple Example of Chaos Control: a 1D Map. The basic idea of controlling chaos can be understood [Lai94] by considering May's classical *logistic map* [May76b]

$$x_{n+1} = f(x_n, r) = rx_n(1 - x_n),$$

where x is restricted to the unit interval $[0, 1]$, and r is a control parameter. It is known that this map develops chaos via the *period-doubling bifurcation* route. For $0 < r < 1$, the asymptotic state of the map (or the attractor of the map) is $x = 0$; for $1 < r < 3$, the attractor is a nonzero fixed-point $x_F = 1 - 1/r$; for $3 < r < 1 + \sqrt{6}$, this fixed-point is unstable and the attractor is a stable period-2 orbit. As r is increased further, a sequence of period-doubling bifurcations occurs in which successive period-doubled orbits become stable. The period-doubling cascade accumulates at $r = r_\infty \approx 3.57$, after which chaos can arise.

Consider the case $r = 3.8$ for which the system is apparently chaotic. An important characteristic of a chaotic attractor is that there exists an infinite number of unstable periodic orbits embedded within it. For example, there are a fixed-point $x_F \approx 0.7368$ and a period-2 orbit with components $x(1) \approx 0.3737$ and $x(2) = 0.8894$, where $x(1) = f(x(2))$ and $x(2) = f(x(1))$.

Now suppose we want to avoid chaos at $r = 3.8$. In particular, we want trajectories resulting from a randomly chosen initial condition x_0 to be as close as possible to the period-2 orbit, assuming that this period-2 orbit gives the best system performance. Of course, we can choose the desired asymptotic state of the map to be any of the infinite number of unstable periodic orbits. Suppose that the parameter r can be finely tuned in a small range around the value $r_0 = 3.8$, i.e., r is allowed to vary in the range $[r_0 - \delta, r_0 + \delta]$, where $\delta \ll 1$. Due to the nature of the chaotic attractor, a trajectory that begins from an arbitrary value of x_0 will fall, with probability one, into the neighborhood of the desired period-2 orbit at some later time. The trajectory would diverge quickly from the period-2 orbit if we do not intervene. Our task is to program the variation of the control parameter so that the trajectory stays in the neighborhood of the period-2 orbit as long as the control is present. In general, the small parameter perturbations will be time dependent [BGL00].

The logistic map in the neighborhood of a periodic orbit can be approximated by a linear equation expanded around the periodic orbit. Denote the target period- m orbit to be controlled as $x(i)$, $i = 1, \dots, m$, where $x(i+1) = f(x(i))$ and $x(m+1) = x(1)$. Assume that at time n , the trajectory falls into the neighborhood of component i of the period- m orbit. The linearized dynamics in the neighborhood of component $i+1$ is then

$$\begin{aligned} x_{n+1} - x(i+1) &= \frac{\partial f}{\partial x}[x_n - x(i)] + \frac{\partial f}{\partial r}\Delta r_n \\ &= r_0[1 - 2x(i)][x_n - x(i)] + x(i)[1 - x(i)]\Delta r_n, \end{aligned}$$

where the partial derivatives are evaluated at $x = x(i)$ and $r = r_0$. We require x_{n+1} to stay in the neighborhood of m . Hence, we set $x_{n+1} - x(i+1) = 0$, which gives

$$\Delta r_n = r_0 \frac{[2x(i) - 1][x_n - x(i)]}{x(i)[1 - x(i)]}. \quad (2.33)$$

Equation (2.33) holds only when the trajectory x_n enters a small neighborhood of the period- m orbit, i.e., when $|x_n - x(i)| \ll 1$, and hence the required parameter perturbation Δr_n is small. Let the length of a small interval defining the neighborhood around each component of the period- m orbit be 2ε . In general, the required maximum parameter perturbation δ is proportional to ε . Since ε can be chosen to be arbitrarily small, δ also can be made arbitrarily small. The average transient time before a trajectory enters the neighborhood of the target periodic orbit depends on ε (or δ). When the trajectory is outside the neighborhood of the target periodic orbit, we do not apply any parameter perturbation, so the system evolves at its nominal parameter value r_0 . Hence we set $\Delta r_n = 0$ when $\Delta r_n > \delta$. The parameter perturbation Δr_n depends on x_n and is time-dependent.

The above strategy for controlling the orbit is very flexible for stabilizing different periodic orbits at different times. Suppose we first stabilize a chaotic trajectory around a period-2 orbit. Then we might wish to stabilize the fixed-point of the logistic map, assuming that the fixed-point would correspond to a better system performance at a later time. To achieve this change of control, we simply turn off the parameter control with respect to the period-2 orbit. Without control, the trajectory will diverge from the period-2 orbit exponentially. We let the system evolve at the parameter value r_0 . Due to the nature of chaos, there comes a time when the chaotic trajectory enters a small neighborhood of the fixed-point. At this time we turn on a new set of parameter perturbations calculated with respect to the fixed-point. The trajectory can then be stabilized around the fixed-point [Lai94].

In the presence of external noise, a controlled trajectory will occasionally be ‘kicked’ out of the neighborhood of the periodic orbit. If this behavior occurs, we turn off the parameter perturbation and let the system evolve by itself. With probability one the chaotic trajectory will enter the neighborhood of the target periodic orbit and be controlled again. The effect of the noise

is to turn a controlled periodic trajectory into an intermittent one in which chaotic phases (uncontrolled trajectories) are interspersed with laminar phases (controlled periodic trajectories). It is easy to verify that the averaged length of the laminar phase increases as the noise amplitude decreases [Lai94].

Floquet Stability Analysis and OGY Control

Controlling chaos, or stabilization of unstable periodic orbits of chaotic systems, has established to a field of large interest since the seed paper of Ott, Grebogi, Yorke [OGY90]. The idea is to stabilize by a feedback calculated at each *Poincaré section*, which reduces the control problem to stabilization of an unstable fixed-point of an iterated map. The feedback can, as in OGY scheme, be chosen proportional to the distance to the desired fixed-point, or proportional to the difference in phase-space position between actual and last but one Poincaré section. This difference control scheme [BDG93], being a time-discrete counterpart of the Pyragas approach [Pyr92, Pyr95], allows for stabilization of inaccurately known fixed-points, and can be extended by a memory term to overcome stability restrictions and to allow for tracking of drifting fixed-points [CMP98a].

In this section the stability of perturbations $x(t)$ around an unstable periodic orbit being subject to a Poincaré-based control scheme is analyzed by means of *Floquet theory* [HL93]. This approach allows to investigate viewpoints that have not been accessible by considering only the iteration dynamics between the Poincaré sections. Among these are primary the discussion of small measurement delays and variable impulse lengths. The impulse length is for both OGY and difference control usually a fixed parameter; and the iterated dynamics is uniquely defined only as long as this impulse length is not varied. The influence of the impulse length has not been point of consideration before; if reported at all, usually for both OGY and difference control a relative length of approximately 1/3 is chosen without any reported sensitivity [Cla02b].

The linearized ODEs of both schemes are invariant under translation in time, $t \rightarrow t+T$. Therefore we can expand the solutions after periodic solutions $u(t+T) = u(t)$ according to

$$x(t) = e^{\gamma T} u_\gamma(t).$$

The necessary condition for stability of the solution is $\text{Re}(\gamma) < 0$; and $x(t) \equiv 0$ refers to motion along the orbit.

Whereas for the *Pyragas control* method (in which the delayed state feedback enforces a time-continuous description) a *Floquet stability analysis* is known [JBO97], here the focus is on the time-discrete control schemes.

Time-Continuous Stability Analysis of OGY Control

Due to the mathematically elegant and practical convenient description and application of OGY control in the Poincaré section up to now there seems

to have been no need to calculate explicitly the *Floquet multiplier* for a stability analysis. However, this allows a novel viewpoint on the differences between the local dynamics around an instable periodic orbit of a dynamical system being subject to Pyragas and OGY control.

For the 1D case, one has the dynamical system [Cla02b]

$$\dot{x}(t) = \lambda x(t) + \mu \varepsilon x(t - (t \bmod T)).$$

In the first time interval between $t = 0$ and $t = T$ the differential equation reads

$$\dot{x}(t) = \lambda x(t) - \mu \varepsilon x(0), \quad \text{for} \quad 0 < t < T.$$

Integration of this differential equation yields

$$x(t) = \left(\left(1 - \frac{\mu \varepsilon}{\lambda}\right) e^{\lambda t} + \frac{\mu \varepsilon}{\lambda} \right) x(0).$$

This gives us an iterated dynamics (here we label the beginning of the time period again with t)

$$x(t + T) = \left(\left(1 - \frac{\mu \varepsilon}{\lambda}\right) e^{\lambda T} + \frac{\mu \varepsilon}{\lambda} \right) x(t).$$

The Floquet multiplier of an orbit therefore is

$$e^{\gamma T} = \left(1 - \frac{\mu \varepsilon}{\lambda}\right) e^{\lambda T} + \frac{\mu \varepsilon}{\lambda}.$$

Influence of the Duration of the Control Impulse on OGY Control

The time-discrete viewpoint now allows to investigate the influence of timing questions on control. First we consider the case that the control impulse is applied timely in the Poincaré section, but only for a finite period $T \cdot p$ within the orbit period ($0 < p < 1$).

This situation is described by the differential equation [Cla02b]

$$\dot{x}(t) = \lambda x(t) + \mu \varepsilon x(t - (t \bmod T)) \cdot \Theta((t \bmod T) - p).$$

Here Θ is a step function ($\Theta(x) = 1$ for $x > 0$ and $\Theta(x) = 0$ elsewhere). In the first time interval between $t = 0$ and $t = T \cdot p$ the differential equation reads

$$\dot{x}(t) = \lambda x(t) + \mu \varepsilon x(0), \quad \text{for} \quad 0 < t < T \cdot p.$$

Integration of this differential equation yields

$$x(t) = \left(\left(1 + \frac{\mu \varepsilon}{\lambda}\right) e^{\lambda t} - \frac{\mu \varepsilon}{\lambda} \right) x(0), \quad x(T \cdot p) = \left(\left(1 + \frac{\mu \varepsilon}{\lambda}\right) e^{\lambda T \cdot p} - \frac{\mu \varepsilon}{\lambda} \right) x(0).$$

In the second interval between $t = T \cdot p$ and $t = T$ the differential equation is the same as without control,

$$\dot{x}(t) = \lambda x(t), \quad \text{for } T \cdot p < t < T.$$

From this one has immediately

$$x(t) = e^{\lambda(t-T \cdot p)} x(T \cdot p).$$

If the beginning of the integration period again is denoted by t , this defines an iteration dynamics,

$$x(t+T) = e^{\lambda(1-p)T} \left(\left(1 + \frac{\mu\varepsilon}{\lambda}\right) e^{\lambda T \cdot p} - \frac{\mu\varepsilon}{\lambda} \right) x(t) = \left(\left(1 + \frac{\mu\varepsilon}{\lambda}\right) e^{\lambda T} - \frac{\mu\varepsilon}{\lambda} e^{\lambda(1-p)T} \right),$$

and the Floquet multiplier of an orbit is given by

$$e^{\gamma T} = \left(1 - \frac{\mu\varepsilon}{\lambda}\right) e^{\lambda T} + \frac{\mu\varepsilon}{\lambda} e^{\lambda(1-p)T} = e^{\lambda T} \left(1 - \frac{\mu\varepsilon}{\lambda} (1 - e^{-\lambda p T})\right). \quad (2.34)$$

One finds that in zero order the ‘strength’ of control is given by the product $p \cdot \mu\varepsilon$; in fact there is a weak linear correction in p . For $\lambda p T \leq 1$ one has

$$e^{\gamma T} = e^{\lambda T} \left(1 + \mu\varepsilon p T + \frac{1}{2} \mu\varepsilon \lambda p^2 T^2 + o(p^3)\right) = e^{\lambda T} \left(1 + \mu\varepsilon p T \left(1 - \frac{1}{2} \lambda p T + o(p^2)\right)\right),$$

i.e., to get a constant strength of control, one has to fulfill the condition

$$\mu\varepsilon p T = \frac{1}{1 - \frac{\lambda T}{2} p} = 1 + \frac{\lambda T}{2} p + o(p^2).$$

The result is, apart from a weak linear correction for OGY control the length of the impulse can be chosen arbitrarily, and the ‘strength’ of control in zero order is given by the time integral over the control impulse.

Blind Chaos Control

One of the most surprising successes of chaos theory has been in biology: the experimentally demonstrated ability to control the timing of spikes of electrical activity in complex and apparently chaotic systems such as heart tissue [GSD92] and brain tissue [SJD94]. In these experiments, PPF control, a modified formulation of OGY control [OGY90], was applied to set the timing of external stimuli; the controlled system showed stable periodic trajectories instead of the irregular inter-spike intervals seen in the uncontrolled system. The mechanism of control in these experiments was interpreted originally as analogous to that of OGY control: unstable periodic orbits riddle the chaotic attractor and the electrical stimuli place the system’s state on the stable manifold of one of these periodic orbits [Kap00].

Alternative possible mechanisms for the experimental observations have been described by Zeng and Glass [GZ94] and Christini and Collins [CC95]. These authors point out that the controlling external stimuli serve to truncate the inter-spike interval to a maximum value. When applied, the control stimulus sets the next interval s_{n+1} to be on the line

$$s_{n+1} = As_n + C. \quad (2.35)$$

We will call this relationship the ‘control line.’ Zeng and Glass showed that if the uncontrolled relationship between inter-spike intervals is a chaotic 1D function, $s_{n+1} = f(s_n)$, then the control system effectively flattens the top of this map and the controlled dynamics may have fixed points or other periodic orbits [GZ94]. Christini and Collins showed that behavior analogous to the fixed-point control seen in the biological experiments can be accomplished even in completely random systems [CC95]. Since neither chaotic 1D systems nor random systems have a stable manifold, the interval-truncation interpretation of the biological experiments is different than the OGY interpretation. The interval-truncation method differs also from OGY and related control methods in that the perturbing control input is a fixed-size stimulus whose timing can be treated as a continuous parameter. This type of input is conventional in cardiology (e.g., [HCT97]).

Kaplan demonstrated in [Kap00] that the state-truncation interpretation was applicable in cases where there was a stable manifold of a periodic orbit as well as in cases where there were only unstable manifolds. He found that superior control could be achieved by intentionally placing the system’s state off of any stable manifold. That suggested a powerful scheme for the rapid experimental identification of fixed points and other periodic orbits in systems where inter-spike intervals were of interest.

The chaos control in [GSD92] and [SJD94] was implemented in two stages. First, inter-spike intervals s_n from the uncontrolled, ‘natural’ system were observed. Modeling the system as a function of two variables

$$s_{n+1} = f(s_n, s_{n-1}),$$

the location s^* of a putative unstable flip-saddle type fixed-point and the corresponding stable eigenvalue λ_s were estimated from the data⁷ [CK97]. The linear approximation to the stable manifold lies on a line given by (2.35) with

$$A = \lambda_s \quad \text{and} \quad C = (1 - \lambda_s)s^*.$$

Second, using estimated values of A and C , the control system was turned on. Following each observed interval s_n , the maximum allowed value of the next inter-spike interval was computed as

$$S_{n+1} = As_n + C.$$

If the next interval naturally was shorter than S_{n+1} no control stimulus was applied to the system. Otherwise, an external stimulus was provided to truncate the inter-spike interval at $s_{n+1} = S_{n+1}$.

In practice, the values of s^* and λ_s for a real fixed-point of the natural system are known only imperfectly from the data. Insofar as the estimates

⁷ Since the fixed-point is unstable, there is also an unstable eigenvalue λ_u .

are inaccurate, the control system does not place the state on the true stable manifold. Therefore, we will analyze the controlled system without presuming that A and C in (2.35) correspond to the stable manifold.

If the natural dynamics of the system is modeled by

$$s_{n+1} = f(s_n, s_{n-1}),$$

then the dynamics of the controlled system is given by [Kap00]

$$s_{n+1} = \min \begin{cases} f(s_n, s_{n-1}) & : \text{Natural Dynamics,} \\ As_n + C & : \text{Control Line.} \end{cases} \quad (2.36)$$

We can study the dynamics of the controlled system close to a natural fixed-point, s^* , by approximating the natural dynamics linearly as⁸

$$s_{n+1} = f(s_n, s_{n-1}) = (\lambda_s + \lambda_u)s_n - \lambda_s\lambda_us_{n-1} + s^*(1 + \lambda_s\lambda_u - \lambda_s - \lambda_u).$$

Since the controlled system (2.36) is nonlinear even when $f()$ is linear, it is difficult to analyze its behavior by algebraic iteration. Nonetheless, the controlled system can be studied in terms of 1D maps.

Following any inter-spike interval when the controlling stimulus has been applied, the system's state (s_n, s_{n-1}) will lie somewhere on the control line. From this time onward the state will lie on an image of the control line even if additional stimuli are applied during future inter-spike intervals.

The stability of the controlled dynamics fixed-point and the size of its basin of attraction can be analyzed in terms of the control line and its image. When the previous inter-spike interval has been terminated by a control stimulus, the state lies somewhere on the control line. If the controlled dynamics are to have a stable fixed-point, this must be at the controller fixed-point x^* where the control line intersects the line of identity. However, the controller fixed-point need not be a fixed-point of the controlled dynamics. For example, if the image of the controller fixed-point is below the controller fixed-point, then the inter-spike interval following a stimulus will be terminated naturally.

For the controller fixed-point to be a fixed-point of the controlled dynamics, we require that the natural image of the controller fixed-point be at or above the controller fixed-point. Thus the dynamics of the controlled system, close to x^* , are given simply by

$$s_{n+1} = As_n + C \quad (2.37)$$

The fixed-point of these dynamics is stable so long as $-1 < A < 1$. In the case of a flip saddle, we therefore have a simple recipe for successful state-truncation control: position x^* below the natural fixed-point s^* and set $-1 < A < 1$.

⁸ Equation (2.37) is simply the linear equation $s_{n+1} = as_n + bs_{n-1} + c$ with a , b , and c set to give eigenvalues λ_s and λ_u and fixed-point s^* .

Type of FP	λ_u	λ_s	x^* Locat.
Flip saddle	$\lambda_u < -1$	$-1 < \lambda_s < 1$	$x^* < s^*$
Saddle	$\lambda_u > 1$	$-1 < \lambda_s < 1$	$x^* > s^*$
Single-flip repeller	$\lambda_u > 1$	$\lambda_s < -1$	$x^* > s^*$
Double-flip repeller	$\lambda_u < -1$	$\lambda_s < -1$	$x^* < s^*$
Spiral (complex λ)	$ \lambda_u > 1$	$ \lambda_s > 1$	$x^* < s^*$

Table 2.1. Cases which lead to a stable fixed-point for the controlled dynamics. In all cases, it is assumed that $|\mathcal{A}| < 1$. (For the cases where $\lambda_s < -1$, the subscript s in λ_s is misleading in that the corresponding manifold is unstable. For the spiral, there is no stable manifold (adapted from [Kap00]).)

Fixed points of the controlled dynamics can exist for natural dynamics other than flip saddles. This can be seen using the following reasoning: Let ξ be the difference between the controller fixed-point and the natural fixed-point: $s^* = x^* + \xi$. Then the natural image of the controller fixed-point can be found from (2.37) to be [Kap00]

$$s_{n+1} = (\lambda_s + \lambda_u)x^* - \lambda_s\lambda_u x^* + (1 + \lambda_s\lambda_u - \lambda_s - \lambda_u)(x^* + \xi). \quad (2.38)$$

The condition that

$$s_{n+1} \geq x^* \quad (2.39)$$

will be satisfied depending only on λ_s , λ_u , and $\xi = s^* - x^*$. This means that for any flip saddle, so long as $x^* < s^*$, the point x^* will be a fixed-point of the controlled dynamics and will be stable so long as $-1 < A < 1$.

Equations (2.38) and (2.39) imply that control can lead to a stable fixed-point for any type of fixed-point except those for which both λ_u and λ_s are greater than 1 (so long as $-1 < A < 1$). Since the required relationship between x^* and s^* for a stable fixed-point of the controlled dynamics depends on the eigenvalues, it is convenient to divide the fixed points into four classes, as given in Table 2.1.

Beyond the issue of the stability of the fixed-point of the controlled dynamics, there is the question of the size of the fixed-point's basin of attraction. Although the local stability of the fixed-point is guaranteed for the cases in Table 2.1 for $-1 < A < 1$, the basin of attraction of this fixed-point may be small or large depending on A , C , s^* , λ_u and λ_s .

The endpoints of the basin of attraction can be derived analytically [Kap00]. The size of the basin of attraction will often be zero when A and C are chosen to match the stable manifold of the natural system. Therefore, in order to make the basin large, it is advantageous intentionally to misplace the control line and to put x^* in the direction indicated in Table 2.1. In addition, control may be enhanced by setting $A \neq \lambda_s$, for instance $A = 0$.

If the relationship between x^* and s^* is reversed from that given in Table 2.1, the controlled dynamics will not have a stable fixed points. To some

extent, these can also be studied using 1D maps. The flip saddle and double-flip repeller can display stable period-2 orbits and chaos. For the non-flip saddle and single-flip repeller, control is unstable when $x^* < s^*$.

The fact that control may be successful or even enhanced when A and C are not matched to λ_s and s^* suggests that it may be useful to reverse the experimental procedure often followed in chaos control. Rather than first identifying the parameters of the natural unstable fixed points and then applying the control, one can blindly attempt control and then deduce the natural dynamics from the behavior of the controlled system. This use of PPF control is reminiscent of pioneering studies that used periodic stimulation to demonstrate the complex dynamics of biological preparations [GGS81].

As an example, consider the *Hénon map*:

$$s_{n+1} = 1.4 + 0.3s_{n-1} - s_n^2.$$

This system has two distinct fixed points. There is a flip-saddle at $s^* = 0.884$ with $\lambda_u = -1.924$ and $\lambda_s = 0.156$ and a non-flip saddle at $s^* = -1.584$ with $\lambda_u = 3.26$ and $\lambda_s = -0.092$. In addition, there is an unstable flip-saddle orbit of period 2 following the sequence $1.366 \rightarrow -0.666 \rightarrow 1.366$. There are no real orbits of period 3, but there is an unstable orbit of period 4 following the sequence $.893 \rightarrow .305 \rightarrow 1.575 \rightarrow -.989 \rightarrow .893$. These facts can be deduced by algebraic analysis of the equations.

In an experiment using the controlled system, the control parameter $x^* = C/(1 - A)$ can be varied. The theory presented above indicates that the controlled system should undergo a bifurcation as x^* passes through s^* . For each value of x^* , the controlled system was iterated from a random initial condition and the values of s_n plotted after allowing a transient to decay. A bifurcation from a stable fixed-point to a stable period 2 as x^* passes through the flip-saddle value of $s^* = 0.884$. A different type bifurcation occurs at the non-flip saddle fixed-point at $s^* = -1.584$. To the left of the bifurcation point, the iterates are diverging to $-\infty$ and are not plotted.

Adding gaussian dynamical noise (of standard deviation 0.05) does not substantially alter the bifurcation diagram, suggesting that examination of the truncation control bifurcation diagram may be a practical way to read off the location of the unstable fixed points in an experimental preparation.

Unstable periodic orbits can be difficult to find in uncontrolled dynamics because there is typically little data near such orbits. Application of PPF control, even blindly, can stabilize such orbits and dramatically improve the ability to locate them. This, and the robustness of the control, may prove particularly useful in biological experiments where orbits may drift in time as the properties of the system change [Kap00].

Jerk Functions of Simple Chaotic Flows

Recall that the celebrated *Lorenz equations* can be written as

$$\dot{x} = -ax + ay, \quad \dot{y} = -xz + bx - y, \quad \dot{z} = xy - cz. \quad (2.40)$$

Note that there are seven terms in the phase-flow of these equations, two of which are nonlinear (xz and xy); also, there are three parameters, for which Lorenz found chaos with $a = 10$, $b = 28$, and $c = 8/3$. The number of independent parameters is generally $d + 1$ less than the number of terms for a d -dimensional system, since each of the variables (x , y , and z in this case) and time (t) can be arbitrarily re-scaled [SL00]. The Lorenz system has been extensively studied, and there is an entire book [Spa82] devoted to it.

Although the Lorenz system is often taken as the prototypical chaotic flow, it is not the algebraically simplest such system [SL00]. Recall that in 1976, O. Rössler [Ros76] proposed his equations,

$$\dot{x} = -y - z, \quad \dot{y} = x + ay, \quad \dot{z} = b + xz - cz. \quad (2.41)$$

Rössler phase-flow also has seven terms and two parameters, which Rössler took as $a = b = 0.2$ and $b = 5.7$, but only a single quadratic nonlinearity (xz).

In 1994, Sprott [Spr94] embarked on an extensive search for autonomous three-dimensional chaotic systems with fewer than seven terms and a single quadratic nonlinearity and systems with fewer than six terms and two quadratic nonlinearities. The *brute-force* method [Spr93a, Spr93b] involved the numerical solution of a huge number (about 10^8) systems of autonomous ODEs with randomly chosen real coefficients and initial conditions. The criterion for chaos was the existence of a leading Lyapunov exponent. He found fourteen algebraically distinct cases with six terms and one nonlinearity, and five cases with five terms and two nonlinearities. One case was volume conserving (conservative), and all the others were volume-contracting (dissipative), implying the existence of a strange attractor. Sprott provided a table of the spectrum of Lyapunov exponents, the related Kaplan-Yorke dimension, and the types and eigenvalues of the unstable fixed-points for each of the nineteen cases [SL00].

Subsequently, Hoover [Hoo95] pointed out that the conservative case found by Sprott in [Spr94]

$$\dot{x} = y, \quad \dot{y} = -x + yz, \quad \dot{z} = 1 - y^2, \quad (2.42)$$

is a special case of the Nosé-Hoover thermostated dynamic system that had earlier been shown [PHV86] to exhibit time-reversible *Hamiltonian chaos*.

In response to Sprott's work, Gottlieb [Got96] pointed out that the conservative system (2.42) could be recast in the *explicit third-order form*

$$\ddot{x} = -\dot{x} + \ddot{x}(x + \ddot{x})/\dot{x},$$

which he called a '*jerk function*' since it involves a third derivative of $\ddot{x}$, which in a mechanical system is the time rate of change of the acceleration, also called the 'jerk' [Sch78]. It is known that any explicit ODE can be cast in the form of a system of coupled first-order ODEs, but the converse does

not hold in general. Even if one can reduce the dynamical system to a jerk form for each of the phase-space variables, the resulting differential equations may look quite different. Gottlieb asked the provocative question ‘What is the simplest jerk function that gives chaos?’

One response was provided by Linz [Lin97] who showed that both the original Rössler model and the Lorenz model can be reduced to jerk forms. The Rössler model (2.41) can be rewritten (in a slightly modified form) as

$$\ddot{x} + (c - \varepsilon + \varepsilon x - \dot{x})\ddot{x} + [1 - \varepsilon c - (1 + \varepsilon^2)x + \varepsilon\dot{x}]\dot{x} + (\varepsilon x + c)x + \varepsilon = 0,$$

where $\varepsilon = 0.2$ and $c = 5.7$ gives chaos. Note that the jerk form of the Rössler equation is a rather complicated quadratic polynomial with 10 terms.

The Lorenz model in (2.40) can be written as

$$\ddot{x} + (1 + \sigma + b - \dot{x}/x)\ddot{x} + [b(1 + \sigma + x^2) - (1 + \sigma)\dot{x}/x]\dot{x} - b\sigma(r - 1 - x^2)x = 0.$$

The jerk form of the Lorenz equation is not a polynomial since it contains terms proportional to $\dot{x}/x$ as is typical of dynamical systems with multiple nonlinearities. Its jerk form contains eight terms.

Linz [Lin97] showed that Sprott’s case R model (see [Spr94]) can be written as a polynomial with only five terms and a single quadratic nonlinearity

$$\ddot{x} + \ddot{x} - x\dot{x} + ax + b = 0,$$

with chaos for $a = 0.9$ and $b = 0.4$.

Sprott [Spr97] also took up Gottlieb’s challenge and embarked on an extensive numerical search for chaos in systems of the explicit form

$$\ddot{x} = J(\ddot{x}, \dot{x}, x),$$

where the jerk function J is a simple quadratic or cubic polynomial. He found a variety of cases, including two with three terms and two quadratic nonlinearities in their jerk function,

$$\ddot{x} + ax\ddot{x} - \dot{x}^2 + x = 0,$$

with $a = 0.645$ and

$$\ddot{x} + ax\ddot{x} - x\dot{x} + x = 0,$$

with $a = -0.113$, and a particularly simple case with three terms and a single quadratic nonlinearity

$$\ddot{x} + a\ddot{x} \pm \dot{x}^2 + x = 0, \tag{2.43}$$

with $a = 2.017$. For this value of a , the Lyapunov exponents are (0.0550, 0, -2.0720) and the Kaplan–Yorke dimension is $d_{KY} = 2.0265$.

Equation (2.43) is simpler than any previously discovered case. The range of the parameter a over which chaos occurs is quite narrow ($2.0168 \dots < a < 2.0577 \dots$). It also has a relatively small basin of attraction, so that initial

conditions must be chosen carefully. One choice of initial conditions that lies in the basin of attraction is $(x, y, z) = (0, 0, \pm 1)$, where the sign is chosen according to the sign of the $\pm \dot{x}^2$ term.

All above systems share a common *route to chaos*. The control parameter a can be considered a *damping rate* for the nonlinear oscillator. For large values of a , there are one or more stable equilibrium points. As a decreases, a *Hopf bifurcation* (see [CD98]) occurs in which the equilibrium becomes unstable, and a stable limit cycle is born. The limit cycle grows in size until it bifurcates into a more complicated limit cycle with two loops, which then bifurcates into four loops, and so forth, in a *sequence of period doublings* until chaos finally onsets. A further decrease in a causes the *chaotic attractor* to grow in size, passing through infinitely many periodic windows, and finally becoming unbounded when the attractor grows to touch the boundary of its *basin of attraction* (a ‘crisis’).

Recently, Malasoma [Mal00] joined the search for simple chaotic jerk functions and found a cubic case as simple as (2.43) but of a different form

$$\ddot{x} + a\dot{x} - x\dot{x}^2 + x = 0,$$

which exhibits chaos for $a = 2.05$. For this value of a , the Lyapunov exponents are $(0.0541, 0, -2.1041)$, and the Kaplan–Yorke dimension is $d_{KY} = 2.0257$. This case follows the usual period–doubling route to chaos, culminating in a boundary crisis and unbounded solutions as a is lowered. The range of a over which chaos occurs is very narrow, $(2.0278 \dots < a < 2.0840 \dots)$. There is also a second extraordinarily small window of chaos for $(0.0753514 \dots < a < 0.0753624 \dots)$, which is five thousand times smaller than the previous case. Malasoma points out that this system is invariant under the parity transformation $x \rightarrow -x$ and speculates that this system is the simplest such example.

Both Linz and Sprott pointed out that if the jerk function is considered the time derivative of an acceleration of a particle of mass m , then *Newton’s second law* implies a *force* F whose time derivative is $\dot{F} = mJ$. If the force has an explicit dependence on only $\dot{x}$, x , and time, it is considered to be ‘Newtonian jerky’. The condition for $F = F(\dot{x}, x, t)$ is that J depends only linearly on $\ddot{x}$. In such a case the force in general includes a *memory term* of the form

$$M = \int_0^t G(x(\tau)) d\tau,$$

which depends on the dynamical history of the motion.

The jerk papers by Linz [Lin97] and Sprott [Spr97] appeared in the same issue of the American Journal of Physics and prompted von Baeyer [Bae98] to comment: “The articles with those funny titles are not only perfectly serious, but they also illustrate in a particularly vivid way the revolution that is transforming the ancient study of mechanics into a new science – one that is not just narrowly concerned with the motion of physical bodies, but that deals with changes of all kinds.” He goes on to say that the method of searching

for chaos in a large class of systems “is not just empty mathematical formalism. Rather it illustrates the arrival of a new level of abstraction in physical science... At that higher level of abstraction, dynamics has returned to the classical Aristotelian goal of trying to understand all change.”

2.1.3 Fractal Calculus and Chaos Control

It is well known that chaos cannot occur in continuous systems of total order less than three. This assertion is based on the usual concepts of order, such as the number of states in a system or the total number of separate differentiations or integrations in the system. The model of system can be rearranged to three single differential equations, where one of the equations contains the non-integer (fractal) order derivative. The total order of system is changed from 3 to $2 + q$, where $0 < q \leq 1$. To put this fact into context, we can consider the fractal-order dynamical model of the system. Hartley *et al.* [HLQ95] consider the fractal-order Chua’s system (compare with an ordinary *Chua’s circuit* (2.15) described in Introduction) and demonstrated that chaos is possible for systems where the order is less than three. In their work, the limits on the mathematical order of the system to have a chaotic response, as measured from the bifurcation diagrams, are approximately from 2.5 to 3.8. In work [ACF98], chaos was discovered in fractal-order two-cell cellular neural networks and also in work [NE99] chaos was exhibited in a system with total order less than three.

The control of chaos has been studied and observed in experiments (e.g., works [Bai89], [LO97], [PY97], [Ush99]). Especially, the control of the well-known Chua’s system [PC89] by sampled data has been studied [YC98]. The main motivation for the control of chaos via sampled data is well-developed digital control techniques.

A sampled-data feedback control of a fractal-order chaotic dynamical system was presented in [Pet99, Pet02], modeled by the state equation $\dot{x} = f(x)$, where $x \in \mathbb{R}^n$ is the state variable, $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is a nonlinear function and $f(0) = 0$. The approach used was concentrating on the feedback control of the chaotic fractal-order Chua’s system, where total order of the system is 2.9.

Fractal Derivatives

Recall that the idea of fractal calculus has been known since the development of the regular calculus, with the first reference probably being associated with Leibniz and L’Hospital in 1695.

Fractal calculus is a generalization of integration and differentiation techniques to a non-integer-order fundamental operator ${}_a D_t^\alpha$, where a and t are the limits of the operation. This continuous integro-differential operator is defined as [Pet99, Pet02]

$${}_aD_t^\alpha = \begin{cases} \frac{d^\alpha}{dt^\alpha} & \text{if } \alpha > 0, \\ 1 & \text{if } \alpha = 0, \\ \int_a^t (d\tau)^{-\alpha} & \text{if } \alpha < 0. \end{cases}$$

The two definitions used for the general fractal integro-differential operator ${}_aD_t^\alpha$ are the *Grünwald-Letnikov definition* (GL) and the *Riemann-Liouville definition* (RL) (see [OS74, Pod99]). The GL definition is given as

$${}_aD_t^\alpha f(t) = \lim_{h \rightarrow 0} h^{-\alpha} \sum_{j=0}^{\lfloor \frac{t-a}{h} \rfloor} (-1)^j \binom{\alpha}{j} f(t-jh), \quad (2.44)$$

where $\lfloor x \rfloor$ means the integer part of x . The RL definition is given as

$${}_aD_t^\alpha f(t) = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dt^n} \int_a^t \frac{f(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau, \quad (2.45)$$

for $(n-1 < \alpha < n)$ and where $\Gamma(\cdot)$ is the *Gamma function*.

For numerical calculation of the fractal-order derivations we can use the relation (2.46) derived from the Grünwald-Letnikov definition (2.44). This approach is based on the fact that for a wide class of functions, the above two definitions, (2.44) and (2.45), are equivalent. The relation for the explicit numerical approximation of the α -th derivative at the points kT , ($k = 1, 2, \dots$) has the following form [Dor94, Pet99, Pod99]

$$({}_{k-L/T}D_{kT}^\alpha f(t) \approx T^{-\alpha} \sum_{j=0}^k (-1)^j \binom{\alpha}{j} f_{k-j}, \quad (2.46)$$

where L is the ‘memory length’, T is the step size of the calculation (sample period), and $(-1)^j \binom{\alpha}{j}$ are binomial coefficients $c_j^{(\alpha)}$, ($j = 0, 1, \dots$). For its calculation we can use

$$c_0^{(\alpha)} = 1, \quad c_j^{(\alpha)} = \left(1 - \frac{1+\alpha}{j}\right) c_{j-1}^{(\alpha)}. \quad (2.47)$$

Now, two general properties of the fractal derivative will be used. The first is the composition of fractal derivative with the integer-order derivative, while the second is the property of linearity.

The fractal-order derivative commutes with the integer-order derivation [Pod99],

$$\frac{d^n}{dt^n} ({}_aD_t^p f(t)) = {}_aD_t^p \left(\frac{d^n f(t)}{dt^n} \right) = {}_aD_t^{p+n} f(t), \quad (2.48)$$

under the condition $t = a$ we have $f^{(k)}(a) = 0$, ($k = 0, 1, 2, \dots, n-1$). The relationship (2.48) says that operators $\frac{d^n}{dt^n}$ and ${}_aD_t^p$ commute.

Similar to the integer-order differentiation, the fractal differentiation is a linear operation [Pod99]:

$${}_aD_t^p (\lambda f(t) + \mu g(t)) = \lambda {}_aD_t^p f(t) + \mu {}_aD_t^p g(t). \quad (2.49)$$

Fractal–Order Chua’s Circuit

Recall that classical Chua’s circuit (2.15) can be written by

$$\begin{aligned}\dot{v}_1 &= \frac{1}{C_1} [G(v_2 - v_1) - f(v_1)], & \dot{v}_2 &= \frac{1}{C_2} [G(v_1 - v_2) + i], \\ \frac{di}{dt} &= \frac{1}{L} [-v_2(t) - R_L i],\end{aligned}\quad (2.50)$$

where $G = 1/R$ and $f(v_1)$ is the piecewise linear $v - i$ characteristic of non-linear Chua’s diode.

Given the above techniques of fractal calculus, there are still a number of ways in which the order of system could be amended. One approach would be to change the order of any or all of three constitutive equations (2.50) so that the total order gives the desired value.

In our case, in the equation one, we replace the first differentiation by the fractal differentiation of order q , ($q \in \mathbb{R}$). The final dimensionless equations of the system for $R_L = 0$ are ($x_1 = v_1, x_2 = v_2, x_3 = i$) [Pet02]

$$\begin{aligned}\dot{x}_1(t) &= \alpha {}_0D_t^{1-q} (x_2(t) - x_1(t) - f(x_1)), & (2.51) \\ \dot{x}_2(t) &= x_1(t) - x_2(t) + x_3(t), \\ \dot{x}_3(t) &= -\beta x_2(t), & \text{where} \\ f(x_1) &= bx_1 + \frac{1}{2}(a - b)(|x_1 + 1| - |x_1 - 1|) \\ \text{with } \alpha &= C_2/C_1, & \beta = C_2 R^2/L.\end{aligned}$$

Feedback Control of Chaos

The structure of the control system with sampled data [YC98] is shown in Figure 2.19. Here, the state variables of the chaotic system are measured and the result is used to construct the output signal $y(t) = Dx(t)$, where D is a constant matrix. The output $y(t)$ is then sampled by sampling block to obtain $y(k) = Dx(k)$ at the discrete moments kT , where $k = 0, 1, 2, \dots$, and T is the sample period. Then $Dx(k)$ is used by the controller to calculate the control signal $u(k)$, which is fed back into chaotic system.

In this way controlled chaotic system is defined by relations [YC98]

$$\begin{aligned}\dot{x}(t) &= f(x(t)) + Bu(k), & t \in [kT, (k+1)T) \\ u(k+1) &= Cu(k) + Dx(k), & (k = 0, 1, 2, \dots)\end{aligned}\quad (2.52)$$

where $u \in \mathbb{R}^m$, $B \in \mathbb{R}^n \times \mathbb{R}^m$, $C \in \mathbb{R}^m \times \mathbb{R}^m$, $D \in \mathbb{R}^m \times \mathbb{R}^n$ and $t \in \mathbb{R}_+$; $x(k)$ is the sampled value of $x(t)$ at $t = kT$. Note that $f(0) = 0$ is an *equilibrium point* of the system (2.52).

The controlled fractal–order Chua’s system is now defined by⁹

⁹ For numerical simulations, the following parameters of the fractal Chua’s system (2.51) were chosen in [Pet02]:

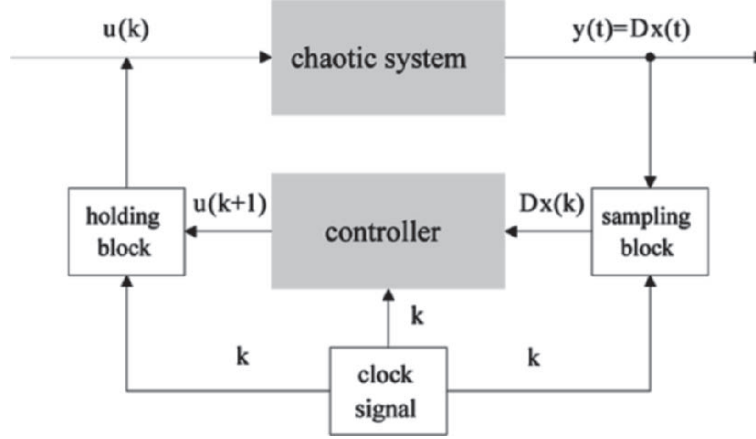


Fig. 2.19. Block-diagram for the chaos-control system (adapted from [YC98]).

$$\begin{aligned}
 \dot{x}_1(t) &= \alpha {}_0D_t^{1-q} (x_2(t) - x_1(t) - f(x_1)) + u_1(t), \\
 \dot{x}_2(t) &= x_1(t) - x_2(t) + x_3(t) + u_2(t), \\
 \dot{x}_3(t) &= -\beta x_2(t) + u_3(t).
 \end{aligned} \tag{2.55}$$

For further details, see [Pet02].

2.1.4 Josephson Junctions and Quantum Computing

Recall that a *Josephson junction* is a type of electronic circuit capable of switching at very high speeds, i.e., frequency of typically $10^{10} - 10^{11}$ Hz, when operated at temperatures approaching absolute zero. It is an insulating barrier separating two superconducting materials and producing the *Josephson*

$$\alpha = 10, \quad \beta = \frac{100}{7}, \quad q = 0.9, \quad a = -1.27, \quad b = -0.68,$$

and the following parameters were experimentally found:

$$B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad C = \begin{pmatrix} 0.8 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad D = \begin{pmatrix} -3.3 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}. \tag{2.53}$$

Using the above parameters (2.53) the digital controller in state space form was defined as

$$u_1(k+1) = 0.8u_1(k) - 3.3x_1(k), \tag{2.54}$$

for $k = 0, 1, 2, \dots$. The initial conditions for Chua's circuit were $((x_1(0), x_2(0), x_3(0)) = (0.2, -0.1, -0.01))$ and the initial condition for the controller (2.54) was $((u_1(0) = (0)))$. The sampling period (frequency) was $T = 100$ Hz. For the computation of the fractal-order derivative in equations (2.55), the relations (2.46), (2.47) and properties (2.48), (2.49) were used. The length of memory was $L = 10$ (1000 coefficients for $T = 100$ Hz).

effect. The terms are named eponymously after British physicist Brian David Josephson, who predicted the existence of the Josephson effect in 1962 [Jos74]. Josephson junction exploits the phenomenon of *superconductivity*, the ability of certain materials to conduct electric current with practically zero resistance. Josephson junctions have important applications in quantum-mechanical circuits. They have great technological promises as amplifiers, voltage standards, detectors, mixers, and fast switching devices for digital circuits. They are used in certain specialized instruments such as highly-sensitive microwave detectors, magnetometers, and QUIDs. Finally, Josephson junctions allow the realization of *qubits*, the key elements of *quantum computers*.

Josephson junctions have been particularly useful for experimental studies of nonlinear dynamics as the equation governing a single junction dynamics is the same as that for a pendulum [Str94]. Their dynamics can be analyzed both in a simple overdamped limit and in the more complex underdamped one, either for single junctions and for arrays of large numbers of coupled junctions.

A Josephson junction is made up of two superconductors, separated by a weak coupling non-superconducting layer, so thin that electrons can cross through the insulating barrier. It can be conceptually represented as:

$$\begin{array}{c} \text{Superconductor 1} : \psi_1 e^{i\phi_1} \\ \text{Weak Coupling} \quad \Updownarrow \\ \text{Superconductor 2} : \psi_2 e^{i\phi_2} \end{array}$$

where the two superconducting regions are characterized by simple *quantum-mechanical wave functions*, $\psi_1 e^{i\phi_1}$ and $\psi_2 e^{i\phi_2}$, respectively. Normally, a much more complicated description would be necessary, as there are $\sim 10^{23}$ electrons to deal with, but in the superconducting ground state, these electrons form the so-called *Cooper pairs* that can be described by a single macroscopic wave function $\psi e^{i\phi}$.

The flow of current between the superconductors in the absence of an applied voltage is called a *Josephson current*, and the movement of electrons across the barrier is known as *Josephson tunnelling*. Two or more junctions joined by superconducting paths form what is called a *Josephson interferometer*.

One of the characteristics of a Josephson junction is that as the temperature is lowered, superconducting current flows through it even in the absence of voltage between the electrodes, part of the *Josephson effect*. The Josephson effect in particular results from two superconductors acting to preserve their long-range order across an insulating barrier. With a thin enough barrier, the phase of the electron wave-function in one superconductor maintains a fixed relationship with the phase of the wave-function in another superconductor. This linking up of phase is called phase coherence. It occurs throughout a single superconductor, and it occurs between the superconductors in a Joseph-

son junction. The *phase coherence*, or *long-range order*, is the essence of the Josephson effect.

The *Josephson-junction quantum computer* was demonstrated in April 1999 by Nakamura, Pashkin and Tsai of NEC Fundamental Research Laboratories in Tsukuba, Japan [NPT99]. In the same month, only about one week earlier, Ioffe, Geshkenbein, Feigel'man, Fauchère and Blatter, independently, described just such a computer in Nature [IGF99].

Nakamura, Pashkin and Tsai's computer is built around a *Cooper pair box*, which is a small superconducting island electrode weakly coupled to a bulk superconductor. Weak coupling between the superconductors creates a Josephson junction between them. Like most other junctions, the Josephson junction is also a capacitor, which is charged by the current that flows through it. A gate voltage is applied between the two superconducting electrodes. If the Cooper box is sufficiently small, e.g., as small as a quantum dot, the charging current breaks into discrete transfer of individual Cooper pairs, so that ultimately it is possible to just transfer a single Cooper pair across the junction. The effectiveness of the Cooper pair transfer depends on the energy difference between the box and the bulk and a maximum is reached when a voltage is applied, which equalizes this energy difference. This leads to *resonance* and observable *coherent quantum oscillations* [Ave99].

This contraption, like the Loss–Vincenzo quantum dot computer [LD98], has the advantage that it is controlled electrically. Unlike Loss–Vincenzo computer, this one actually exists in the laboratory. Nakamura, Pashkin and Tsai did not perform any computations with it though. At this stage it was enough of an art to observe the coherence for about 6 cycles of the Cooper pair oscillations, while the chip was cooled to about and carefully shielded from external electromagnetic radiation.

There are two general types of Josephson junctions: overdamped and underdamped. In overdamped junctions, the barrier is conducting (i.e., it is a normal metal or superconductor bridge). The effects of the junction's internal electrical resistance will be large compared to its small capacitance. An overdamped junction will quickly reach a unique equilibrium state for any given set of conditions.

The barrier of an underdamped junction is an insulator. The effects of the junction's internal resistance will be minimal. Underdamped junctions do not have unique equilibrium states, but are hysteretic.

Driven Pendulum Analog

To show a driven pendulum analog of a microscopic description of a single Josephson junction, we start with:

1. The *Josephson current–phase relation*

$$I = I_c \sin \phi,$$

where I_c is the *critical current*, I is the bias current, and $\phi = \phi_2 - \phi_1$ is the constant *phase difference* between the phases of the two superconductors that are weakly coupled; and

2. The *Josephson voltage–phase relation*

$$V = \frac{\hbar}{2e} \dot{\phi},$$

where $V = V(t)$ is the instantaneous voltage across the junction, $\hbar$ is the Planck constant (divided by 2π), and e is the charge on the electron.

Now, if we apply Kirchoff's voltage and current laws for the parallel RC-circuit with resistance R and capacitance C , we come to the first-order ODE

$$C\dot{V} + \frac{V}{R} + I_c \sin \phi = I,$$

which can be recast solely in terms of the phase difference ϕ as the second-order pendulum-like ODE,

$$\begin{aligned} \text{Josephson junction :} \quad & \frac{\hbar C}{2e} \ddot{\phi} + \frac{\hbar}{2eR} \dot{\phi} + I_c \sin \phi = I, \\ \text{Pendulum :} \quad & ml^2 \ddot{\theta} + b\dot{\theta} + mgl \sin \theta = \tau. \end{aligned} \quad (2.56)$$

This mechanical analog has often proved useful in visualizing the dynamics of Josephson Junctions [Str94]. If we divide (2.56) by I_c and define a dimensionless time

$$\tau = \frac{2eI_c R}{\hbar} t,$$

we get the dimensionless oscillator equation for Josephson junction,

$$\beta \phi'' + \phi' + \sin \phi = \frac{I}{I_c}, \quad (2.57)$$

where $\phi' = d\phi/d\tau$. The dimensionless group β , defined by

$$\beta = \frac{2eI_c R^2 C}{\hbar},$$

is called the McCumber parameter and represents a dimensionless capacitance.

In a simple *overdamped limit* $\beta \ll 1$ with *resistive loading*, the ‘inertial term’ $\beta \phi''$ may be neglected (as if oscillating in a highly-viscous medium), and so (2.57) reduces to a non-uniform oscillator

$$\phi' = \frac{I}{I_c} - \sin \phi, \quad (2.58)$$

with solutions approaching a stable fixed-point for $I < I_c$, and periodically varying for $I > I_c$. To find the current–voltage curve in the overdamped limit,

we take the average voltage $\langle V \rangle$ as a function of the constant applied current I , assuming that all transients have decayed and the system has reached steady-state, and get

$$\langle V \rangle = I_c R \langle \phi' \rangle.$$

An overdamped *array of N Josephson Junctions* (2.58), parallel with a resistive load R , can be described by the system of first-order dimensionless ODEs [Str94]

$$\phi'_k = \Omega + a \sin \phi_k + \frac{1}{N} \sum_{j=1}^N \sin \phi_j, \quad k = 1, \dots, N,$$

where

$$\begin{aligned} \Omega &= I_b R_0 / I_c r, & a &= -(R_0 + r)/r, & R_0 &= R/N, \\ I_b &= I_c \sin \phi_k + \frac{\hbar}{2eR} \dot{\phi}_k + \frac{\hbar}{2eR} \sum_{j=1}^N \dot{\phi}_j. \end{aligned}$$

2.1.5 Solitons

Recall that the story of solitary waves, or *solitons*, started in 1834, when young Scottish engineer named John Scott Russell was conducting experiments on the Union Canal (near Edinburgh) to measure the relationship between the speed of a boat and its propelling force, with the aim of finding design parameters for conversion from horse power to steam. One August day, a rope parted in his measurement apparatus and [Rus844] “The boat suddenly stopped – not so the mass of water in the channel which it had put in motion; it accumulated round the prow of the vessel in a state of violent agitation, then suddenly leaving it behind, rolled forward with great velocity, assuming the form of a large solitary elevation, a rounded, smooth and well defined heap of water, which continued its course along the channel without change of form or diminution of speed.” Russell did not ignore this unexpected phenomenon, but “followed it on horseback, and overtook it still rolling on at a rate of some eight or nine miles an hour, preserving its original figure some thirty feet long and a foot to a foot and a half in height” until the wave became lost in the windings of the channel. He continued to study the solitary wave in tanks and canals over the following decade, finding it to be an independent dynamic entity moving with constant shape and speed.

Using a wave tank he demonstrated four facts [Rus844]:

- (i) Solitary waves have the shape $h \operatorname{sech}^2[k(x - vt)]$;
- (ii) A sufficiently large initial mass of water produces two or more independent solitary waves;
- (iii) Solitary waves cross each other “without change of any kind”;
- (iv) A wave of height h and traveling in a channel of depth d has a velocity given by the expression

$$v = \sqrt{g(d+h)}, \quad (2.59)$$

(where g is the acceleration of gravity), implying that a large amplitude solitary wave travels faster than one of low amplitude.

Although confirmed by observations on the Canal de Bourgogne, near Dijon, most subsequent discussions of the hydrodynamic solitary wave missed the physical significance of Russell's observations. Evidence that to the end of his life Russell maintained a much broader and deeper appreciation of the importance of his discovery is provided by a posthumous work, where he correctly estimated the height of the earth's atmosphere from (2.59) and the fact that "the sound of a cannon travels faster than the command to fire it" [Rus885].

In 1895, D. Korteweg and H. de Vries published a theory of shallow water waves that reduced Russell's problem to its essential features. One of their results was the nonlinear PDE

$$u_t + c u_x + \varepsilon u_{xxx} + \gamma u u_x = 0, \quad (2.60)$$

which would play a key role in soliton theory [KdV95]. In this equation, $u(x, t)$ is the wave amplitude, $c = \sqrt{gd}$ is the speed of small amplitude waves, $\varepsilon \equiv c(d^2/6 - T/2\rho g)$ is a dispersive parameter, $\gamma \equiv 3c/2d$ is a nonlinear parameter, and T and ρ are respectively the surface tension and the density of water. The authors showed that (2.60) has a family of exact traveling wave solutions of the form $u(x, t) = \tilde{u}(x - vt)$, where $\tilde{u}(\cdot)$ is Russell's "rounded, smooth and well defined heap" and v is the wave speed. If the dispersive term (ε) and the nonlinear term (γ) in (2.60) are both zero, then the *Korteweg-de Vries equation* (KdV) becomes linear,

$$u_t + c u_x = 0,$$

with a traveling wave solution for any pulse shape at the fixed speed $v = c = \sqrt{gd}$. In general the KdV equation (2.60) is nonlinear with exact traveling wave solutions

$$u(x, t) = h \operatorname{sech}^2[k(x - vt)], \quad (2.61)$$

where $k = \sqrt{\gamma h/12\varepsilon}$, implying that higher amplitude waves are more narrow. With this shape, the effects of dispersion balance those of nonlinearity at an adjustable value of the pulse speed. Thus the solitary wave is recognized as an independent dynamic entity, maintaining a dynamic balance between these two influences. Interestingly, solitary wave velocities are related to amplitudes by

$$v = c + \gamma h/3 = \sqrt{gd}(1 + h/2d), \quad (2.62)$$

in accord with Russell's empirical results, given in (2.59), to $O(h)$.

Although unrecognized at the time, such an energy conserving solitary wave is related to the existence of a *Bäcklund transform* (BT), which was proposed by J.O. Bäcklund in 1885 [Lam76]. In such a transform, a known

solution generates a new solution through a single integration, after which the new solution can be used to generate another new solution, and so on. It is straightforward to find a BT for any linear PDE, which introduces a new eigenfunction into the total solution with each application of the transform. Only special nonlinear PDEs are found to have BTs, but late nineteenth mathematicians knew that these include

$$u_{\xi\tau} = \sin u, \quad (2.63)$$

which arose in research on the geometry of curved surfaces [Ste36].

In 1939, Y. Frenkel and T. Kontorova introduced a seemingly unrelated problem arising in solid state physics to model the relationship between dislocation dynamics and plastic deformation of a crystal [FK39]. From this study, a PDE describing dislocation motion is

$$u_{xx} - u_{tt} = \sin u, \quad (2.64)$$

where $u(x, t)$ is atomic displacement in the x -direction and the ‘sin’ function represents periodicity of the crystal lattice. A *traveling wave solution* of (2.64), corresponding to the propagation of a dislocation, is

$$u(x, t) = 4 \arctan \left[\exp \left(\frac{x - vt}{\sqrt{1 - v^2}} \right) \right], \quad (2.65)$$

with velocity v in the range $(-1, +1)$. Since (2.64) is identical to (2.63) after an independent variable transformation [$\xi = (x - t)/2$ and $\tau = (x + t)/2$], exact solutions of (2.64) involving arbitrary numbers of dislocation components as in (2.65) can be generated through a succession of Bäcklund transforms, but this was not known to Frenkel and Kontorova.

In the late 1940s, E. Fermi, J. Pasta and S. Ulam (the famous FPU-trio) suggested one of the first scientific problems to be assigned to the Los Alamos MANIAC computing machine: the dynamics of energy equipartition in a slightly nonlinear crystal lattice, which is related to thermal conductivity. The system they chose was a chain of 64 equal mass particles connected by slightly nonlinear springs, so from a linear perspective there were 64 normal modes of oscillation in the system. It was expected that if all the initial energy were put into a single vibrational mode, the small nonlinearity would cause a gradual progress toward equal distribution of the energy among all modes (thermalization), but the numerical results were surprising. If all the energy is originally in the mode of lowest frequency, it returns almost entirely to that mode after a period of interaction among a few other low frequency modes. In the course of several numerical refinements, no thermalization was observed [FPU55].

Pursuit of an explanation for this “FPU recurrence” led Zabusky and Kruskal to approximate the nonlinear spring-mass system by the KdV equation. In 1965, they reported numerical observations that KdV solitary waves

pass through each other with no change in shape or speed, and coined the term “soliton” to suggest this property [ZK65].

N.J. Zabusky and M.D. Kruskal were not the first to observe nondestructive interactions of energy conserving solitary waves. Apart from Russell’s tank measurements [Rus844], Perring and Skyrme, had studied solutions of Equation (2.64) comprising two solutions as in Equation (2.65) undergoing a collision. In 1962, they published numerical results showing perfect recovery of shapes and speeds after a collision and went on to discover an exact analytical description of this phenomenon [PS62].

This result would not have surprised nineteenth century mathematicians; it is merely the second member of the hierarchy of solutions generated by a Bäcklund transform. Nor would it have been unexpected by Seeger and his colleagues, who had noted in 1953 the connections between the nineteenth century work [Ste36] and the studies of Frenkel and Kontorova [SDK53]. Since Perring and Skyrme were interested in Equation (2.64) as a nonlinear model for elementary particles of matter, however, the complete absence of scattering may have been disappointing.

Throughout the 1960s, (2.64) arose in a variety of problems including the propagation of ferromagnetic domain walls, self-induced transparency in nonlinear optics, and the propagation of magnetic flux quanta in long Josephson transmission lines. Eventually (2.64) became known as the *sine-Gordon (SG) equation*, which is a nonlinear version of the linear *Klein-Gordon equation*:

$$u_{xx} - u_{tt} = u.$$

Perhaps the most important contribution made by Zabusky and Kruskal in their 1965 paper was to recognize the relation between nondestructive soliton collisions and the riddle of FPU recurrence. Viewing KdV solitons as independent and localized dynamic entities, they explained the FPU observations as follows. The initial condition generates a family of solitons with different speeds, moving apart in the $x - t$ plane. Since the system studied was of finite length with perfect reflections at both ends, the solitons could not move infinitely far apart; instead they eventually reassembled in the $x - t$ plane, approximately recreating the initial condition after a surprisingly short *recurrence time*.

By 1967, this insight had led Kruskal and his colleagues to devise a nonlinear generalization of the Fourier transform method for constructing solutions of the KdV equation emerging from arbitrary initial conditions [GGK67]. Called the inverse scattering (or inverse spectral) method (ISM), this approach proceeds in three steps.

1. The nonlinear KdV dynamics are mapped onto an associated linear problem, where each eigenvalue of the linear problem corresponds to the speed of a particular KdV soliton.
2. Since the associated problem is linear, the time evolution of its solution is readily computed.

3. An inverse calculation then determines the time evolved KdV dynamics from the evolved solution of the linear associated problem. Thus the solution of a nonlinear problem is found from a series of linear computations.

Another development of the 1960s was M. Toda's discovery of exact two-soliton interactions on a nonlinear spring-mass system, called *Toda lattice* [Tod67]. As in the FPU system, equal masses were assumed to be interconnected with nonlinear springs, but Toda chose the potential

$$\left(\frac{a}{b}\right) [e^{-bu_j} - 1] + au_j, \quad (2.66)$$

where $u_j(t)$ is the longitudinal extension of the j th spring from its equilibrium value and both a and b are adjustable parameters. (In the limit $a \rightarrow \infty$ and $b \rightarrow 0$ with ab finite, this reduces to the quadratic potential of a linear spring. In the limit $a \rightarrow 0$ and $b \rightarrow \infty$ with ab finite, it describes the interaction between hard spheres.) Thus by the late 1960s, it was established – although not widely known – that solitons were not limited to PDEs (KdV and SG). Local solutions of difference-differential equations could also exhibit the unexpected properties of unchanging shapes and speeds after collisions.

These events are only the salient features of a growing panorama of nonlinear wave activities that became gradually less parochial during the 1960s. Solid state physicists began to see relationships between their solitary waves (magnetic domain walls, self-shaping pulses of light, quanta of magnetic flux, polarons, etc.), and those from classical hydrodynamics and oceanography, while applied mathematicians began to suspect that the ISM might be used for a broader class of nonlinear wave equations. It was amid this intellectual ferment that A.C. Newell and his colleagues organized the first soliton research workshop during the summer of 1972 [New74]. Interestingly, one of the most significant contributions to this conference came by post. From the Soviet Union arrived a paper by V.E. Zakharov and A.B. Shabat, formulating Kruskal's ISM for the nonlinear PDE [ZS72]

$$iu_t + u_{xx} + 2|u|^2u = 0. \quad (2.67)$$

In contrast to KdV, SG and the Toda lattice, the dependent variable in this equation is complex rather than real, so the evolutions of two quantities (magnitude and phase of u) are governed by the equation. This reflects the fact that (2.67) is a nonlinear generalization of a linear PDE

$$iu_t + u_{xx} + u = 0,$$

solutions of which comprise both an envelope and a carrier wave. Since this linear equation is a *Schrödinger equation* for the quantum mechanical probability amplitude of a particle (like an electron) moving through a region of uniform potential, it is natural to call (2.67) the *nonlinear Schrödinger (NLS) equation*. When the NLS equation is used to model wave packets in

such fields as hydrodynamics, nonlinear optics, nonlinear acoustics, plasma waves and biomolecular dynamics, however, its solutions are devoid of quantum character.

Upon appreciating the Zakharov and Shabat paper, many left the 1972 conference convinced that four nonlinear equations (KdV, SG, NLS, and the Toda lattice) display solitary wave behavior with the special properties that led Zabusky and Kruskal to coin the term soliton [New74]. Within two years, ISM formulations had been constructed for the SG equation and also for the Toda lattice.

Since the mid-1970s, the soliton concept has become established in several areas of applied science, and dozens of dynamic systems are now known to be integrable through the ISM. Even if a system is not exactly integrable, additionally, it may be close to an integrable system, allowing analytic insight to be gleaned from perturbation theory. Thus one is no longer surprised to find stable spatially localized regions of energy, balancing the opposing effects of nonlinearity and dispersion and displaying the essential properties of objects. In the last two decades, soliton studies have been focused on working out the details of such object-like behavior in a wide range of research areas [Sco99].

Example: Nerve Impulse Propagation

The nerve impulse propagation along the axon of a neuron can be studied by combining the equations for an excitable membrane with the differential equations for an electrical core conductor cable, assuming the axon to be an infinitely long cylinder. A well known approximation of FitzHugh [Fit61] and Nagumo [NAY60] to describe the propagation of voltage pulses $V(x, t)$ along the membranes of nerve cells is the set of coupled PDEs

$$V_{xx} - V_t = F(V) + R - I, \quad R_t = c(V + a - bR), \quad (2.68)$$

where $R(x, t)$ is the recovery variable, I the external stimulus and a, b, c are related to the membrane radius, specific resistivity of the fluid inside the membrane and temperature factor respectively.

When the spatial variation of V , namely V_{xx} , is negligible, (2.68) reduces to the Van der Pol oscillator,

$$\dot{V} = V - \frac{V^3}{3} - R + I, \quad \dot{R} = c(V + a - bR),$$

with $F(V) = -V + \frac{V^3}{3}$. Normally the constants in (2.68) satisfy the inequalities $b < 1$ and $3a + 2b > 3$, though from a purely mathematical point of view this need not be insisted upon. Then with a periodic (ac) applied membrane current $A_1 \cos \omega t$ and a (dc) bias A_0 , the Van der Pol equation becomes

$$\dot{V} = V - \frac{V^3}{3} - R + A_0 + A_1 \cos \omega t, \quad \dot{R} = c(V + a - bR). \quad (2.69)$$

Further, (2.69) can be rewritten as a single second-order ODE by differentiating $\dot{V}$ with respect to time and using $\dot{R}$ for R ,

$$\begin{aligned} \ddot{V} - (1 - bc) \left\{ 1 - \frac{V^2}{1 - bc} \right\} \dot{V} - c(b - 1)V + \frac{bc}{3}V^3 \\ = c(A_0b - a) + A_1 \cos(\omega t + \phi), \end{aligned} \quad (2.70)$$

where $\phi = \tan^{-1} \frac{\omega}{bc}$. Using the transformation $x = (1 - bc)^{-(1/2)}V$, $t \longrightarrow t' = t + \frac{\phi}{\omega}$, (2.70) can be rewritten as

$$\begin{aligned} \ddot{x} + p(x^2 - 1)\dot{x} + \omega_0^2 x + \beta x^3 &= f_0 + f_1 \cos \omega t, \quad \text{where} \quad (2.71) \\ p &= (1 - bc), \quad \omega_0^2 = c(1 - b), \quad \beta = bc \frac{(1 - bc)}{3}, \\ f_0 &= c \frac{(A_0b - a)}{\sqrt{1 - bc}}, \quad f_1 = \frac{A_1}{\sqrt{1 - bc}}. \end{aligned}$$

Note that (2.71), or its rescaled form

$$\ddot{x} + p(kx^2 + g)\dot{x} + \omega_0^2 x + \beta x^3 = f_0 + f_1 \cos \omega t, \quad (2.72)$$

is the *Duffing–Van der Pol equation*. In the limit $k = 0$, we have the Duffing equation discussed below (with $f_0 = 0$), and in the case $\beta = 0$ ($g = -1$, $k = 1$) we have the forced van der Pol equation. Equation (2.72) exhibits a very rich variety of bifurcations and chaos phenomena, including quasi-periodicity, phase lockings and so on, depending on whether the potential $V = \frac{1}{2}\omega_0^2 x^2 + \frac{\beta x^4}{4}$ is i) a double well, ii) a single well or iii) a double hump [Lak97, Lak03].

2.1.6 Chaos Field Theory

In [Cvi00], Cvitanovic re-examined the path-integral formulation and the role that the classical solutions play in quantization of strongly nonlinear fields. In the path integral formulation of a field theory the dominant contributions come from saddle-points, the classical solutions of equations of motion. Usually one imagines *one dominant saddle point*, the ‘vacuum’ (see Figure 2.20, (a)).

The *Feynman diagrams* of quantum electrodynamics (QED) and quantum chromodynamics (QCD), associated to their path integrals, give us a visual and intuitive scheme to calculate the correction terms to this starting semiclassical, *Gaussian saddlepoint approximation*. But there might be other saddles (Figure 2.20, (b)). That field theories might have a rich repertoire of classical solutions became apparent with the discovery of *instantons* [BPS75], analytic solutions of the classical $SU(2)$ *Yang–Mills relation*, and the realization that the associated *instanton vacua* receive contributions from countable ∞ ’s of saddles. What is not clear is whether these are the important classical

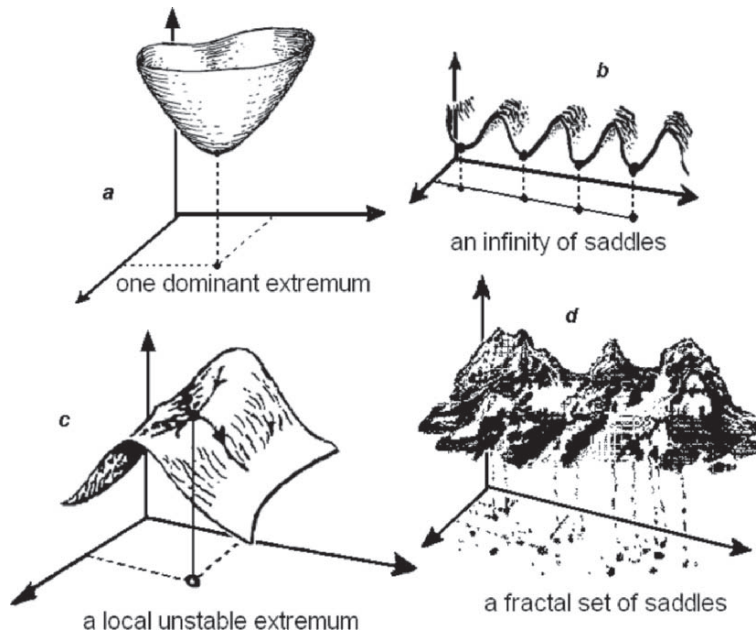


Fig. 2.20. Path integrals and chaos field theory (see text for explanation).

saddles. Cvitanovic asks the question: could it be that the strongly nonlinear theories are dominated by altogether different classical solutions?

The search for the classical solutions of nonlinear field theories such as the *Yang-Mills* and *gravity* has so far been neither very successful nor very systematic. In modern field theories the main emphasis has been on symmetries (compactly collected in action functionals that define the theories) as guiding principles in writing down the actions. But writing down a differential equation is only the start of the story; even for systems as simple as 3 coupled ordinary differential equations one in general has no clue what the nature of the long time solutions might be.

These are hard problems, and in explorations of modern field theories the dynamics tends to be neglected, and understandably so, because the wealth of the classical solutions of nonlinear systems can be truly bewildering. If the classical behavior of these theories is anything like that of the field theories that describe the classical world – the hydrodynamics, the magnetohydrodynamics, the *Burgers dynamical system*, *Ginzburg-Landau equation*, or *Kuramoto-Sivashinsky equation*, there should be very many solutions, with very few of the important ones analytical in form; the strongly nonlinear classical field theories are turbulent, after all. Furthermore, there is not a dimmest hope that such solutions are either beautiful or analytic, and there is not much enthusiasm for grinding out numerical solutions as long as one lacks ideas as what to do with them.

By late 1970's it was generally understood that even the simplest nonlinear systems exhibit chaos. Chaos is the norm also for generic Hamiltonian flows, and for path integrals that implies that instead of a few, or countably few saddles (Figure 2.20, (c)), classical solutions populate fractal sets of saddles (Figure 2.20, (d)). For the path-integral formulation of quantum mechanics such solutions were discovered and accounted for by [Gut90] in late 1960's. In this framework the spectrum of the theory is computed from a set of its unstable classical periodic solutions and quantum corrections. The new aspect is that the individual saddles for classically chaotic systems are nothing like the harmonic oscillator degrees of freedom, the quarks and gluons of QCD – they are all unstable and highly nontrivial, accessible only by numerical techniques.

So, if one is to develop a semiclassical field theory of systems that are *classically chaotic* or *turbulent*, the problem one faces is twofold [Cvi00]

1. Determine, classify, and order by relative importance the classical solutions of nonlinear field theories.
2. Develop methods for calculating perturbative corrections to the corresponding classical saddles.

2.1.7 Geometrical Nonlinear Dynamics

Here we give a paradigm of geometrical modeling and analysis of complex continuous dynamical systems (see [II06b] for technical details). This is essentially a *recipe* how to develop a *covariant formalism on smooth manifolds*, given a certain physical, or bio-physical, or psycho-physical, or socio-physical system, here labeled by a generic name: ‘physical situation’. We present this recipe in the form of the following five-step algorithm.

(I) So let's start: given a certain physical situation, the first step in its predictive modeling and analysis, that is, in applying a powerful differential-geometric machinery to it, is to associate with this situation *two* independent coordinate systems, constituting two independent smooth Riemannian manifolds. Let us denote these two coordinate systems and their respective manifolds as:

- *Internal coordinates:* $x^i = x^i(t)$, ($i = 1, \dots, m$), constituting the m D *internal configuration manifold*: $M^m \equiv \{x^i\}$; and
- *External coordinates:* $y^e = y^e(t)$, ($e = 1, \dots, n$), constituting the n D *external configuration manifold*: $N^n \equiv \{y^e\}$.

The main example that we have in mind is a standard robotic or bio-dynamic (loco)motion system, in which x^i denote internal joint coordinates, while y^e denote external Cartesian coordinates of segmental centers of mass. However, we believe that such developed methodology can fit a generic physical situation.

Therefore, in this first, engineering step (I) of our differential-geometric modeling, we associate to the given natural system, not one but two different

and independent smooth configuration manifolds, somewhat like viewing from two different satellites a certain place on Earth with a football game playing in it.

(II) Once that we have precisely defined two smooth manifolds, as two independent views on the given physical situation, we can apply our differential-geometric modeling to it and give it a natural physical interpretation. More precisely, once we have two smooth Riemannian manifolds, $M^m \equiv \{x^i\}$ and $N^n \equiv \{y^e\}$, we can formulate two smooth maps between them:¹⁰

$$f : N \rightarrow M, \text{ given by coordinate transformation: } x^i = f^i(y^e), \quad (2.73)$$

and

$$g : M \rightarrow N, \text{ given by coordinate transformation: } y^e = g^e(x^i). \quad (2.74)$$

If the Jacobian matrices of these two maps are nonsingular (regular), that is if their Jacobian determinants are nonzero, then these two maps are mutually inverse, $f = g^{-1}$, and they represent standard *forward and inverse kinematics*.

(III) Although, maps f and g define some completely general nonlinear coordinate (functional) transformations, which are even unknown at the moment, there is something linear and simple that we know about them (from calculus). Namely, the corresponding infinitesimal transformations are linear and homogenous: from (2.73) we have (applying everywhere Einstein's summation convention over repeated indices)

$$dx^i = \frac{\partial f^i}{\partial y^e} dy^e, \quad (2.75)$$

while from (2.74) we have

$$dy^e = \frac{\partial g^e}{\partial x^i} dx^i. \quad (2.76)$$

Furthermore, (2.75) implies the linear and homogenous transformation of *internal velocities*,

$$v^i \equiv \dot{x}^i = \frac{\partial f^i}{\partial y^e} \dot{y}^e, \quad (2.77)$$

while (2.76) implies the linear and homogenous transformation of *external velocities*,

$$u^e \equiv \dot{y}^e = \frac{\partial g^e}{\partial x^i} \dot{x}^i. \quad (2.78)$$

In this way, we have defined *two velocity vector-fields*, the internal one: $v^i = v^i(x^i, t)$ and the external one: $u^e = u^e(y^e, t)$, given respectively by the two nonlinear systems of ODEs, (2.77) and (2.78).¹¹

¹⁰ This obviously means that we are working in the category of smooth manifolds.

¹¹ Although transformations of differentials and associated velocities are linear and homogeneous, the systems of ODE's define nonlinear vector-fields, as they include Jacobian (functional) matrices.

(IV) The next step in our differential-geometrical modeling/analysis is to define second derivatives of the manifold maps f and g , that is the two *acceleration vector-fields*, which we will denote by $a^i = a^i(x^i, \dot{x}^i, t)$ and $w^e = w^e(y^e, \dot{y}^e, t)$, respectively. However, unlike simple physics in linear Euclidean spaces, these two acceleration vector-fields on manifolds M and N are not the simple time derivatives of the corresponding velocity vector-fields ($a^i \neq \dot{v}^i$ and $w^e \neq \dot{u}^e$), due to the existence of the *Levi-Civita connections* ∇_M and ∇_N on both M and N . Properly defined, these two acceleration vector-fields respectively read:

$$a^i = \dot{v}^i + \Gamma_{jk}^i v^j v^k = \ddot{x}^i + \Gamma_{jk}^i \dot{x}^j \dot{x}^k, \quad \text{and} \quad (2.79)$$

$$w^e = \dot{u}^e + \Gamma_{hl}^e u^h u^l = \ddot{y}^e + \Gamma_{hl}^e \dot{y}^h \dot{y}^l, \quad (2.80)$$

where Γ_{jk}^i and Γ_{hl}^e denote the (second-order) *Christoffel symbols* of the connections ∇_M and ∇_N .

Therefore, in the step (III) we gave the first-level model of our physical situation in the form of two ordinary vector-fields, the first-order vector-fields (2.77) and (2.78). For some simple situations (e.g., modeling ecological systems), we could stop at this modeling level. Using physical terminology we call them velocity vector-fields. Following this, in the step (IV) we have defined the two second-order vector-fields (2.79) and (2.80), as a connection-base derivations of the previously defined first-order vector-fields. Using physical terminology, we call them ‘acceleration vector-fields’.

(V) Finally, following our generic physical terminology, as a natural next step we would expect to define some kind of generic Newton–Maxwell force-fields. And we can actually do this, with a little surprise that individual forces involved in the two force-fields will not be vectors, but rather the dual objects called 1-forms (or, 1D differential forms). Formally, we define the two *covariant force-fields* as

$$F_i = m g_{ij} a^j = m g_{ij} (\dot{v}^j + \Gamma_{ik}^j v^i v^k) = m g_{ij} (\ddot{x}^j + \Gamma_{ik}^j \dot{x}^i \dot{x}^k), \quad \text{and} \quad (2.81)$$

$$G_e = m g_{eh} w^h = m g_{eh} (\dot{u}^h + \Gamma_{el}^h u^e u^l) = m g_{eh} (\ddot{y}^h + \Gamma_{el}^h \dot{y}^e \dot{y}^l), \quad (2.82)$$

where m is the mass of each single segment (unique, for simplicity), while $g_{ij} = g_{ij}^M$ and $g_{eh} = g_{eh}^N$ are the two *Riemannian metric tensors* corresponding to the manifolds M and N . The two force-fields, F_i defined by (2.81) and G_e defined by (2.82), are generic force-fields corresponding to the manifolds M and N , which represent the *material cause* for the given physical situation. Recall that they can be physical, bio-physical, psycho-physical or socio-physical force-fields. Physically speaking, they are the *generators* of the corresponding dynamics and kinematics.

Main geometrical relations behind this fundamental paradigm, forming the *covariant force functor* [II06b], were given in diagram 1.3 in Introduction.

2.1.8 Dynamics of n –Functors

Recall from [II06b] that *dynamics of n –functors* is given by the following

Recursive Lagrangian Homotopy Dynamics:

0-cell : $x_0 \bullet \quad x_0 \in M; \quad$ in the higher cells below: $t, s \in [0, 1];$

1-cell : $x_0 \bullet \xrightarrow{f} \bullet x_1 \quad f : x_0 \simeq x_1 \in M,$
 $f : [0, 1] \rightarrow M, f : x_0 \mapsto x_1, x_1 = f(x_0), f(0) = x_0, f(1) = x_1;$

e.g., linear path: $f(t) = (1-t)x_0 + tx_1; \quad$ or

e.g., Euler-Lagrangian f -dynamics with endpoint conditions $(x_0, x_1) :$

$$\frac{d}{dt}f_{\dot{x}^i} = f_{x^i}, \quad \text{with} \quad x(0) = x_0, \quad x(1) = x_1, \quad (i = 1, \dots, n);$$

$$2\text{-cell} : x_0 \bullet \begin{array}{c} \xrightarrow{f} \\ \Downarrow h \\ \xrightarrow{g} \end{array} \bullet x_1 \quad h : f \simeq g \in M,$$

$h : [0, 1] \times [0, 1] \rightarrow M, h : f \mapsto g, g = h(f(x_0)),$

$h(x_0, 0) = f(x_0), h(x_0, 1) = g(x_0), h(0, t) = x_0, h(1, t) = x_1$

e.g., linear homotopy: $h(x_0, t) = (1-t)f(x_0) + tg(x_0); \quad$ or

e.g., homotopy between two Euler-Lagrangian (f, g) -dynamics
 with the same endpoint conditions $(x_0, x_1) :$

$$\frac{d}{dt}f_{\dot{x}^i} = f_{x^i}, \quad \text{and} \quad \frac{d}{dt}g_{\dot{x}^i} = g_{x^i} \quad \text{with} \quad x(0) = x_0, \quad x(1) = x_1;$$

$$3\text{-cell} : x_0 \bullet \begin{array}{c} \xrightarrow{f} \\ \left(\begin{array}{c} \Downarrow h \\ \Downarrow j \\ \Downarrow i \\ \Downarrow g \end{array} \right) \\ \xrightarrow{g} \end{array} \bullet x_1 \quad j : h \simeq i \in M,$$

$j : [0, 1] \times [0, 1] \times [0, 1] \rightarrow M, j : h \mapsto i, i = j(h(f(x_0)))$

$j(x_0, t, 0) = h(f(x_0)), j(x_0, t, 1) = i(f(x_0)),$

$j(x_0, 0, s) = f(x_0), j(x_0, 1, s) = g(x_0),$

$j(0, t, s) = x_0, j(1, t, s) = x_1$

e.g., linear composite homotopy: $j(x_0, t, s) = (1-t)h(f(x_0)) + ti(f(x_0));$
 or, homotopy between two homotopies between above two Euler-Lagrangian
 (f, g) -dynamics with the same endpoint conditions (x_0, x_1) .

2.2 Systems and Networks Approach

In this section we give a brief review on control and complex systems and networks approach to brain and cognition. Formally, here we are dealing with a *system evolution* 2–functor $\mathcal{E}$, given by

$$\begin{array}{ccc}
 \begin{array}{ccc}
 A & \xrightarrow{f} & B \\
 \downarrow h & \text{CURRENT} & \downarrow g \\
 & \text{SYSTEM} & \\
 \downarrow & \text{STATE} & \downarrow \\
 C & \xrightarrow{k} & D
 \end{array} & \xrightarrow{\mathcal{E}} & \begin{array}{ccc}
 \mathcal{E}(A) & \xrightarrow{\mathcal{E}(f)} & \mathcal{E}(B) \\
 \downarrow \mathcal{E}(h) & \text{DESIRED} & \downarrow \mathcal{E}(g) \\
 & \text{SYSTEM} & \\
 \downarrow & \text{STATE} & \downarrow \\
 \mathcal{E}(C) & \xrightarrow{\mathcal{E}(k)} & \mathcal{E}(D)
 \end{array}
 \end{array}$$

Here $\mathcal{E}$ represents an association/projection functor from the *source 2–category* of the current system state, defined as a commutative square of small system categories $A, B, C, D, \dots$ of *current system components* and their causal interrelations $f, g, h, k, \dots$, onto the *target 2–category* of the desired system state, defined as a commutative square of small system categories $\mathcal{E}(A), \mathcal{E}(B), \mathcal{E}(C), \mathcal{E}(D), \dots$ of *evolved system components* and their causal interrelations $\mathcal{E}(f), \mathcal{E}(g), \mathcal{E}(h), \mathcal{E}(k)$. As in the previous section, each causal arrow in above diagram, e.g., $f : A \rightarrow B$, stands for a *generic system dynamorphism*.

2.2.1 MIMO Systems and Kalman Filtering

Basics of Kalman’s Linear State–Space Theory

It is well-known that linear multiple input–multiple output (MIMO) control systems can always be put into Kalman canonical state–space form of order n , with m inputs and k outputs. In the case of *continual time systems* we have the state and output equations of the form¹²

$$\begin{aligned}
 \dot{\mathbf{x}} &= \mathbf{A}(t) \mathbf{x}(t) + \mathbf{B}(t) \mathbf{u}(t), \\
 \mathbf{y}(t) &= \mathbf{C}(t) \mathbf{x}(t) + \mathbf{D}(t) \mathbf{u}(t),
 \end{aligned} \tag{2.83}$$

while in case of *discrete time systems* we have the state and output equations of the form

$$\begin{aligned}
 \mathbf{x}(n+1) &= \mathbf{A}(n) \mathbf{x}(n) + \mathbf{B}(n) \mathbf{u}(n), \\
 \mathbf{y}(n) &= \mathbf{C}(n) \mathbf{x}(n) + \mathbf{D}(n) \mathbf{u}(n).
 \end{aligned} \tag{2.84}$$

¹² In the covariant form, (2.84) reads [II06b]

$$\dot{x}^i = a_j^i x^j + b_k^i u^k, \quad y^i = c_j^i x^j + d_k^i u^k, \quad (i, j = 1, \dots, n; \quad k = 1, \dots, m).$$

Both in (2.83) and in (2.84) the variables have the following meaning:

$\mathbf{x}(t) \in \mathbb{X}$ is an n -vector of *state variables* belonging to the *state space* $\mathbb{X} \subset \mathbb{R}^n$;

$\mathbf{u}(t) \in \mathbb{U}$ is an m -vector of *inputs* belonging to the *input space* $\mathbb{U} \subset \mathbb{R}^m$;

$\mathbf{y}(t) \in \mathbb{Y}$ is a k -vector of *outputs* belonging to the *output space* $\mathbb{Y} \subset \mathbb{R}^k$;

$\mathbf{A}(t) : \mathbb{X} \rightarrow \mathbb{X}$ is an $n \times n$ matrix of *state dynamics*;

$\mathbf{B}(t) : \mathbb{U} \rightarrow \mathbb{X}$ is an $n \times m$ matrix of *input map*;

$\mathbf{C}(t) : \mathbb{X} \rightarrow \mathbb{Y}$ is an $k \times n$ matrix of *output map*;

$\mathbf{D}(t) : \mathbb{U} \rightarrow \mathbb{Y}$ is an $k \times m$ matrix of *input-output transform*.

Input $\mathbf{u}(t) \in \mathbb{U}$ can be empirically determined by trial and error; it is properly defined by optimization process called *Kalman regulator*, or more generally (in the presence of noise), by *Kalman filter* (even better, *extended Kalman filter* to deal with stochastic nonlinearities) [Kal60].

Now, the most common special case of the general Kalman model (2.83), with constant state, input and output matrices (and relaxed boldface vector-matrix notation), is the so-called *stationary linear model*. Such systems frequently serve as a baseline, against which other control systems are measured. We follow a common notational convention and let u denote the vector of *inputs*, y the vector of *outputs* and assume that they can be related through an intermediary *state* variable x according to the equations

$$\dot{x} = Ax + Bu, \quad y = Cx. \quad (2.85)$$

We refer to this as the deterministic *stationary linear model*. The stationary linear system (2.85) defines a variety of operators, in particular those related to: (i) regulators, (ii) end point controls, (iii) servomechanisms, and (iv) repetitive modes (see [Bro01]).

The discrete-time version of (2.85) is described in the next subsection.

Basics of Kalman Filtering

The classical *Kalman filter* is an effective procedure for *combining noisy sensor outputs to estimate the state* of a *system with uncertain dynamics*. The Kalman filter provides a *recursive solution* to the *linear optimal filtering problem*. It applies to stationary as well as non-stationary environments. The solution is recursive in that each updated estimate of the state is computed from the previous estimate and the new input data, so only the previous estimate requires storage. In addition to eliminating the need for storing the entire past observed data, the Kalman filter is computationally more efficient than computing the estimate directly from the entire past observed data at each step of the filtering process.

Recall from section 6.3 above, that the *linear quadratic regulator* (LQR) represents a *linear state feedback control law*

$$u = -Kx$$

for the linear MIMO-system

$$\dot{x} = Ax + Bu$$

which minimizes a *quadratic cost function*

$$J = \int_0^\infty (x(t)^T Q x(t) + u(t)^T R u(t)) dt.$$

The control law is called *optimal with respect to the cost function J*.

Now, one might ask whether there is an optimal design technique for a *state estimator*. That is, is there an approach to *observer design* which is equivalent, in some sense, to the linear quadratic regulator?

Given the *observable system*

$$\dot{x} = Ax + Bu, \quad y = Cx,$$

one may define the *dual system*

$$\dot{\theta} = A^T \theta + C^T \gamma$$

and design an LQR controller to minimize the quadratic cost function

$$J = \int_0^\infty (\theta(t)^T Q \theta(t) + \gamma(t)^T R \gamma(t)) dt.$$

However, it is unclear how one should ‘penalize’ θ and γ in the cost function. Instead, consider the *extended observable system*

$$\dot{x} = Ax + Bu + w, \quad y = Cx + v,$$

in which the dynamics are subject to random disturbances w and the measurements are subject to random noise v . In parallel with the development of the linear quadratic regulator, Rudolph Kalman examined the following *optimal estimator problem*: Construct a full state observer which minimizes the combined effect of the disturbances and the noise, thus providing a ‘most likely’ estimate of the system state. Solving this problem requires some information about the random processes. If the processes are zero-mean, Gaussian white noise processes, then the optimal estimator design problem becomes perfectly analogous to the LQR control design problem. In 1960, Kalman published his famous paper describing a recursive solution to the discrete-data linear filtering problem [Kal60]. Since that time, due in large part to advances in digital computing, the Kalman filter has been the subject of extensive particularly in the area of autonomous or assisted navigation (see, e.g., [Hay01, GWA01]).

The Kalman filter is a *discrete-time, two-step process*, the steps of which are usually called *prediction* and *correction*, thus resembling a popular *Adams–Bashforth–Moulton integrator* for ODEs (see, e.g., [WB95]). The *predictor*, or *time update*, projects the current system’s state estimate ahead in time. The

corrector, or *measurement update*, adjusts the projected state estimate by an actual system's measurement at that time. In this way, the correction step makes corrections to an estimate, based on new information obtained from sensor measurements. The continuous-time version is usually referred to as *Kalman-Bucy filter* or *smoother* [SL03, Asi04].

Consider a generic linear, discrete-time dynamical system. The concept of *discrete state* is fundamental to this description. The *state vector*, denoted by x_k , is defined as the minimal set of data that is sufficient to uniquely describe the unforced dynamical behavior of the system; the subscript k denotes discrete time. In other words, the state is the least amount of data on the past behavior of the system that is needed to predict its future behavior. Typically, the state x_k is unknown. To estimate it, we use a set of observed data, denoted by the *observable* vector y_k .

The state-space model of a generic linear, discrete-time dynamical system includes the *process equation* (3.120) and the *measurement equation* (3.36)

$$x_{k+1} = F_{k+1,k} x_k + w_k, \quad (2.86)$$

$$y_k = H_k x_k + v_k, \quad (2.87)$$

where F_{k+1} is the *transition matrix* taking the state x_k from time k to time $k+1$, H_k is the *measurement sensitivity matrix*, while w_k and v_k are independent, additive, zero-mean, white *Gaussian noise* processes, defined below.

The covariance matrix of the *process noise* w_k is defined by

$$E[w_n, w_k^T] = \begin{cases} Q_k, & \text{for } n = k, \\ 0, & \text{for } n \neq k. \end{cases}$$

Similarly, the covariance matrix of the *measurement noise* v_k is defined by

$$E[v_n, v_k^T] = \begin{cases} R_k, & \text{for } n = k, \\ 0, & \text{for } n \neq k. \end{cases}$$

The *Kalman filtering problem*, namely, the problem of jointly solving the process and measurement equations for the unknown state in an optimum manner may now be formally stated as follows: Use the entire observed data, consisting of the vectors $y_1, y_2, \dots, y_k$, to find for each $k \geq 1$ the minimum mean-square error estimate of the state x_i . The problem is called *filtering* if $i = k$, *prediction* if $i > k$, and *smoothing* if $1 \leq i < k$.

The derivation of the Kalman filter is based on the following two theorems (see [Kal60, Hay01]):

- *Conditional mean estimator.* If the stochastic processes $\{x_k\}$ and $\{y_k\}$ are jointly Gaussian, then the optimum estimate $\hat{x}_k$ that minimizes the mean-square error J_k is the conditional mean estimator:

$$\hat{x}_k = E[x_k | y_1, y_2, \dots, y_k].$$

- *Principle of orthogonality.* Let the stochastic processes $\{x_k\}$ and $\{y_k\}$ be of zero means; that is,

$$E[x_k] = E[y_k] = 0, \quad \text{for all } k.$$

Then:

- (i) the stochastic process $\{x_k\}$ and $\{y_k\}$ are jointly Gaussian; or
- (ii) if the optimal estimate $\hat{x}_k$ is restricted to be a linear function of the observables and the cost function is the mean-square error,
- (iii) then the optimum estimate $\hat{x}_k$, given the observables $y_1, y_2, \dots, y_k$, is the orthogonal projection of x_k on the space spanned by these observables.

The *Kalman filter design algorithm* consists of (see [Kal60, Hay01]):

1. *Initialization:* For $k = 0$, set

$$\hat{x}_0 = E[x_0], \quad P_0 = E[(x_0 - E[x_0])(x_0 - E[x_0])^T].$$

and

2. *Computation:* For $k = 1, 2, \dots$, compute:

- (i) *State estimate propagation*

$$\hat{x}_{\bar{k}} = F_{k,k-1} \hat{x}_{\bar{k}-1};$$

- (ii) *Error covariance propagation*

$$P_{\bar{k}} = F_{k,k-1} P_{k-1} F_{k,k-1}^T + Q_{k-1};$$

- (iii) *Kalman gain matrix*

$$G_k = P_{\bar{k}} H_k^T [H_k P_{\bar{k}} H_k^T + R_k]^{-1};$$

- (iv) *State estimate update*

$$\hat{x}_k = \hat{x}_{\bar{k}} + G_k (y_k - H_k \hat{x}_{\bar{k}});$$

- (v) *Error covariance update*

$$P_k = (I - G_k H_k) P_{\bar{k}}.$$

Therefore, the basic Kalman filter is a linear, discrete-time, finite-dimensional system, which is endowed with a recursive structure that makes a digital computer well suited for its implementation. A key property of the Kalman filter is that it is the minimum mean-square (variance) estimator of the state of a linear dynamical system. The model is stochastic owing to the additive presence of process noise and measurement noise, which are assumed to be Gaussian with zero mean and known covariance matrices.

Extended Kalman Filter

The Kalman filtering problem considered so far has addressed the estimation of a state vector in a linear model of a dynamical system. If, however, the model is nonlinear, we may extend the use of Kalman filtering through a linearization procedure. The resulting filter is referred to as the *extended Kalman filter* (EKF) [Hay01]. Such an extension is feasible by virtue of the fact that the Kalman filter is described in terms of difference equations in the case of discrete-time systems. While the ordinary (i.e., linear) Kalman filter is defined in terms of the measurement sensitivity matrix H_k , the extended Kalman filter can be defined in terms of a suitably differentiable vector-valued measurement sensitivity function $h(k, x_k)$.

To set the stage for a development of the extended Kalman filter, consider a nonlinear dynamical system described by the state-space model

$$x_{k+1} = f(k, x_k) + w_k, \quad y_k = h(k, x_k) + v_k, \quad (2.88)$$

where, as before, w_k and v_k are independent zero-mean white Gaussian noise processes with covariance matrices R_k and Q_k , respectively. Here, however, the functional $f(k, x_k)$ denotes a nonlinear transition matrix function that is possibly time-variant. Likewise, the functional $h(k, x_k)$ denotes a vector-valued measurement sensitivity function, i.e., a nonlinear measurement matrix that may be time-variant, too [Hay01].

The basic idea of the extended Kalman filter is to linearize the state-space model (2.88) at each time instant around the most recent state estimate, which is taken to be either $\hat{x}_k$ or $\hat{x}_{\bar{k}}$, depending on which particular functional is being considered. Once a linear model is obtained, the standard Kalman filter equations are applied.

The *EKF design algorithm* consists of [Hay01]:

1. The *discrete state-space model* (2.88).
2. *Definitions*

$$F_{k,k} = \left. \frac{\partial f(k, x)}{\partial x} \right|_{x=x_k}, \quad H_k = \left. \frac{\partial h(k, x)}{\partial x} \right|_{x=x_{\bar{k}}}.$$

3. *Initialization*: For $k = 0$, set

$$\hat{x}_0 = E[x_0], \quad P_0 = E[(x_0 - E[x_0])(x_0 - E[x_0])^T].$$

4. *Computation*: For $k = 1, 2, \dots$, compute:
 - (i) *State estimate propagation*

$$\hat{x}_{\bar{k}} = F_{k,k-1} \hat{x}_{\bar{k}-1};$$

- (ii) *Error covariance propagation*

$$P_{\bar{k}} = F_{k,k-1} P_{k-1} F_{k,k-1}^T + Q_{k-1};$$

(iii) *Kalman gain matrix*

$$G_k = P_{\bar{k}} H_k^T [H_k P_{\bar{k}} H_k^T + R_k]^{-1};$$

(iv) *State estimate update*

$$\hat{x}_k = \hat{x}_{\bar{k}} + G_k (y_k - H_k \hat{x}_{\bar{k}});$$

(v) *Error covariance update*

$$P_k = (I - G_k H_k) P_{\bar{k}}.$$

Sensor Fusion in Hybrid Systems

Kalman filter can be used to *combine* or *fuse* information from different sensors for hybrid systems (see Figure 2.21), like accelerometers and gyroscopes (see text below). The basic idea is to use the Kalman filter to weight the different mediums most heavily in the circumstances where they each perform best, thus providing more accurate and stable estimates than a system based on any one medium alone (see [Lui02]). In particular, the *indirect feedback Kalman filter* (also called a *complementary* or *error-state Kalman filter*) is often used to combine the two mediums [May79]. In such a configuration, the Kalman filter is used to estimate the difference between the current inertial and optical (or acoustic) outputs, i.e., it continually estimates the error in the inertial estimates by using the optical system as a second (redundant) reference. This error estimate is then used to correct the inertial estimates. The *tuning* of the Kalman filter parameters then adjusts the weight of the correction as a function of frequency. By slightly modifying the Kalman filter, adaptive velocity response can be incorporated also. This can be accomplished by adjusting (in real time) the expected optical measurement error as a function of the magnitude of velocity [WB95]. Kalman filter has been used to investigate the human balancing system [VKD01, VKD03] (see below).

2.2.2 Nonlinear Control Systems

In this subsection we give a brief introduction to *geometrical nonlinear control systems*. Majority of techniques developed under this name consider the so-called *affine nonlinear MIMO-systems* of the form (see [Isi89, NS90, Lew95, LM97])

$$\dot{x}(t) = f_0(x(t)) + u^i(t) f_i(x(t)), \quad (i = 1, \dots, m) \quad (2.89)$$

where $t \mapsto x(t)$ is a curve in a system's *state manifold* M . The vector-field f_0 is called the *drift vector-field*, describing the dynamics of the system in the absence of controls, and the vector-fields $f_1, \dots, f_m$ are the *input vector-fields* or *control vector-fields*, indicating how we are able to actuate the system. The vector-fields $f_0, f_1, \dots, f_m$ are assumed to be real analytic. We do not ask for any sort of linear independence of the control vector-fields $f_1, \dots, f_m$. We shall

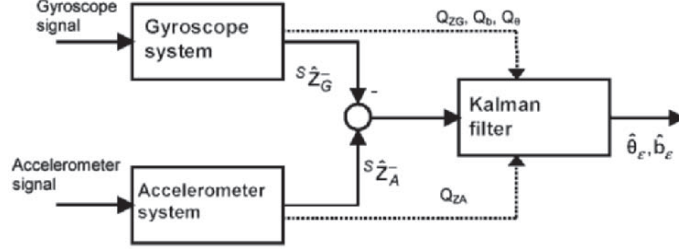


Fig. 2.21. Structure of *sensor fusion in Kalman filter estimation*. Both the accelerometer and the gyroscope system are used to make an estimate of the global vertical unit vector $\mathbf{Z}$. The difference between the two estimates is written as a function of orientation error θ_ϵ and offset error $\mathbf{b}_\epsilon$. Kalman filter estimates both θ_ϵ and $\mathbf{b}_\epsilon$ using the error covariances of the orientation $\mathbf{Q}_\theta$, offset $\mathbf{Q}_b$ and inclination estimation ($\mathbf{Q}_{ZG}$ and $\mathbf{Q}_{ZA}$). These estimated errors are used to correct the estimated.

suppose that the controls $u : [0, T] \rightarrow U$ are locally integrable with U some subset of $\mathbb{R}^m$. We allow the length T of the interval on which the control is defined to be arbitrary. It is convenient to denote by $\tau(u)$ the right endpoint of the interval for a given control u . For a fixed U we denote by $\mathcal{U}$ the collection of all measurable controls taking their values in U . To be concise about this, a *control affine system* is a triple $\Sigma = (M, \mathcal{F} = \{f_0, f_1, \dots, f_m\}, U)$, with all objects as defined above. A *controlled trajectory* for Σ is a pair (c, u) , where $u \in \mathcal{U}$ and where $c : [0, \tau(u)] \rightarrow M$ is defined so that

$$\dot{c}(t) = f_0(c(t)) + u^i(t)f_i(c(t)).$$

One can show that for admissible controls, the curve c will exist at least for sufficiently small times, and that the initial condition $c(0) = x_0$ uniquely defines c on its domain of definition.

For $x \in M$ and $T > 0$ we define several types of *reachable sets* as:

$$\begin{aligned} \mathcal{R}_\Sigma(x, T) &= \{c(T) : \\ &(c, u) \text{ is a controlled trajectory for } \Sigma \text{ with } \tau(u) = T \text{ and } c(0) = x\}, \\ \mathcal{R}_\Sigma(x, \leq T) &= \cup_{t \in [0, T]} \mathcal{R}_\Sigma(x, t), \quad \mathcal{R}_\Sigma(x) = \cup_{t \geq 0} \mathcal{R}_\Sigma(x, t), \end{aligned}$$

that allow us to give several definitions of controllability as follows. Let $\Sigma = (M, \mathcal{F}, U)$ be a control affine system and let $x \in M$. We say that:

1. Σ is *accessible* from x if $\text{int}(\mathcal{R}_\Sigma(x)) \neq \emptyset$.
2. Σ is *strongly accessible* from x if $\text{int}(\mathcal{R}_\Sigma(x, T)) \neq \emptyset$ for each $T > 0$.
3. Σ is *locally controllable* from x if $x \in \text{int}(\mathcal{R}_\Sigma(x))$.
4. Σ is *small-time locally controllable* (STLC) from x if there exists $T > 0$ so that $x \in \text{int}(\mathcal{R}_\Sigma(x, \leq T))$ for each $t \in [0, T]$.
5. Σ is *globally controllable* from x if $(\mathcal{R}_\Sigma(x)) = M$.

For example, a typical simple system that is accessible but not controllable is given by the following data:

$$M = \mathbb{R}^2, \quad m = 1, \quad U = [-1, 1],$$

$$\dot{x} = u, \quad \dot{y} = x^2.$$

This system is (not obviously) accessible from $(0, 0)$, but is (obviously) not locally controllable from that same point. Note that although $\mathcal{R}_\Sigma((0, 0), \leq T)$ has nonempty interior, the initial point $(0, 0)$ is not in that interior. Thus this is a system that is not controllable in any sense. Note that the system is also strongly accessible.

Feedback Linearization

Recall that the core of control theory is the idea of the *feedback*. In case of nonlinear control, this implies *feedback linearization*.

Exact Feedback Linearization

The idea of feedback linearization is to algebraically transform the nonlinear system dynamics into a fully or partly linear one so that the linear control techniques can be applied. Note that this is not the same as a conventional linearization using Jacobians. In this subsection we will present the modern, geometrical, Lie-derivative based techniques for exact feedback linearization of nonlinear control systems.

The Lie Derivative and Lie Bracket in Control Theory. Given a scalar function $h(x)$ and a vector-field $f(x)$, we define a new scalar function, $\mathcal{L}_f h = \nabla h f$, which is the Lie derivative of h w.r.t. f , i.e., the directional derivative of h along the direction of the vector f . Repeated Lie derivatives can be defined recursively:

$$\mathcal{L}_f^0 h = h, \quad \mathcal{L}_f^i h = \mathcal{L}_f \left(\mathcal{L}_f^{i-1} h \right) = \nabla \left(\mathcal{L}_f^{i-1} h \right) f, \quad (\text{for } i = 1, 2, \dots)$$

Or given another vector-field, g , then $\mathcal{L}_g \mathcal{L}_f h(x)$ is defined as

$$\mathcal{L}_g \mathcal{L}_f h = \nabla (\mathcal{L}_f h) g.$$

For example, if we have a control system

$$\dot{x} = f(x), \quad y = h(x),$$

with the state $x = x(t)$ and the output y , then the derivatives of the output are:

$$\dot{y} = \frac{\partial h}{\partial x} \dot{x} = \mathcal{L}_f h, \quad \text{and} \quad \ddot{y} = \frac{\partial \mathcal{L}_f h}{\partial x} \dot{x} = \mathcal{L}_f^2 h.$$

Also, recall that the curvature of two vector-fields, g_1, g_2 , gives a non-zero Lie bracket, $[g_1, g_2]$ see Figure 2.22). Lie bracket motions can generate new directions in which the system can move.

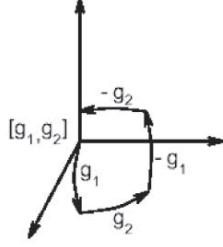


Fig. 2.22. The so-called ‘Lie bracket motion’ is possible by appropriately modulating the control inputs (see text for explanation).

In general, the Lie bracket of two vector-fields, $f(x)$ and $g(x)$, is defined by

$$[f, g] = Ad_f g = \nabla g f - \nabla f g = \frac{\partial g}{\partial x} f - \frac{\partial f}{\partial x} g,$$

where $\nabla f = \partial f / \partial x$ is the Jacobian matrix. We can define Lie brackets recursively,

$$Ad_f^0 g = g, \quad Ad_f^i g = [f, Ad_f^{i-1} g], \quad (\text{for } i = 1, 2, \dots)$$

Lie brackets have the properties of bilinearity, skew-commutativity and Jacobi identity.

For example, if

$$f = \begin{pmatrix} \cos x_2 \\ x_1 \end{pmatrix}, \quad g = \begin{pmatrix} x_1 \\ 1 \end{pmatrix},$$

then we have

$$[f, g] = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \cos x_2 \\ x_1 \end{pmatrix} - \begin{pmatrix} 0 & -\sin x_2 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ 1 \end{pmatrix} = \begin{pmatrix} \cos x_2 + \sin x_2 \\ -x_1 \end{pmatrix}.$$

Input/Output Linearization. Given a single-input single-output (SISO) system

$$\dot{x} = f(x) + g(x)u, \quad y = h(x), \quad (2.90)$$

we want to formulate a linear-ODE relation between output y and a new input v . We will investigate (see [Isi89, SI89, Wil00]):

- How to generate a linear input/output relation.
- What are the internal dynamics and zero-dynamics associated with the input/output linearization?
- How to design stable controllers based on the I/O linearization.

This linearization method will be exact in a finite domain, rather than tangent as in the local linearization methods, which use Taylor series approximation. Nonlinear controller design using the technique is called exact feedback linearization.

Algorithm for Exact Feedback Linearization. We want to find a nonlinear compensator such that the closed-loop system is linear (see Figure 2.23). We will consider only affine SISO systems of the type (2.90), i.e., $\dot{x} = f(x) + g(x)u$, $y = h(x)$, and we will try to construct a *control law* of the form

$$u = p(x) + q(x)v, \quad (2.91)$$

where v is the setpoint, such that the *closed-loop nonlinear system*

$$\dot{x} = f(x) + g(x)p(x) + g(x)q(x)v, \quad y = h(x),$$

is linear from command v to y .

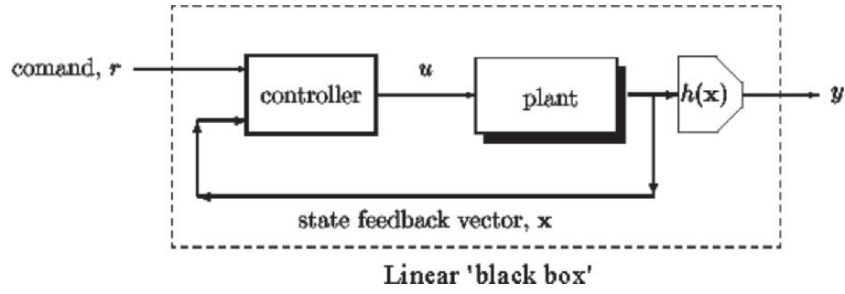


Fig. 2.23. Feedback linearization (see text for explanation).

The main idea behind the feedback linearization construction is to find a nonlinear change of coordinates which transforms the original system into one which is linear and controllable, in particular, a chain of integrators. The difficulty is finding the output function $h(x)$ which makes this construction possible.

We want to design an exact nonlinear feedback controller. Given the nonlinear affine system, $\dot{x} = f(x) + g(x)u$, $y = h(x)$, we want to find the controller functions $p(x)$ and $q(x)$. The unknown functions inside our controller (2.91) are given by:

$$p(x) = \frac{-\left(\mathcal{L}_f^r h(x) + \beta_1 \mathcal{L}_f^{r-1} h(x) + \dots + \beta_{r-1} \mathcal{L}_f h(x) + \beta_r h(x)\right)}{\mathcal{L}_g \mathcal{L}_f^{r-1} h(x)},$$

$$q(x) = \frac{1}{\mathcal{L}_g \mathcal{L}_f^{r-1} h(x)}, \quad (2.92)$$

which are comprised of Lie derivatives, $\mathcal{L}_f h(x)$. Here, the *relative order*, r , is the smallest integer r such that $\mathcal{L}_g \mathcal{L}_f^{r-1} h(x) \neq 0$. For linear systems r is the difference between the number of poles and zeros.

To get the *desired response*, we choose the r parameters in the β polynomial to describe how the output will respond to the setpoint, v (pole-placement).

$$\frac{d^r y}{dt^r} + \beta_1 \frac{d^{r-1} y}{dt^{r-1}} + \dots + \beta_{r-1} \frac{dy}{dt} + \beta_r y = v.$$

Here is the proposed algorithm [Isi89, SI89, Wil00]):

1. Given nonlinear SISO process, $\dot{x} = f(x, u)$, and output equation $y = h(x)$, then:
2. Calculate the relative order, r .
3. Choose an r th order desired linear response using pole-placement technique (i.e., select β). For this could be used a simple r th order low-pass filter such as a Butterworth filter.
4. Construct the exact linearized nonlinear controller (2.92), using Lie derivatives and perhaps a symbolic manipulator (Mathematica or Maple).
5. Close the loop and get a linear input-output black-box (see Figure 2.23).
6. Verify that the result is actually linear by comparing with the desired response.

Relative Degree

A nonlinear SISO system

$$\dot{x} = f(x) + g(x)u, \quad y = h(x),$$

is said to have *relative degree* r at a point x_o if (see [Isi89, NS90])

1. $L_g L_f^k h(x) = 0$ for all x in a neighborhood of x_o and all $k < r - 1$; and
2. $L_g L_f^{r-1} h(x_o) \neq 0$.

For example, *controlled Van der Pol oscillator* has the state-space form

$$\dot{x} = f(x) + g(x)u = \begin{bmatrix} x_2 \\ 2\omega\zeta(1 - \mu x_1^2)x_2 - \omega^2 x_1 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u.$$

Suppose the output function is chosen as $y = h(x) = x_1$. In this case we have

$$\begin{aligned} L_g h(x) &= \frac{\partial h}{\partial x} g(x) = [1 \ 0] \begin{bmatrix} 0 \\ 1 \end{bmatrix} = 0, \quad \text{and} \\ L_f h(x) &= \frac{\partial h}{\partial x} f(x) = [1 \ 0] \begin{bmatrix} x_2 \\ 2\omega\zeta(1 - \mu x_1^2)x_2 - \omega^2 x_1 \end{bmatrix} = x_2. \end{aligned}$$

Moreover

$$L_g L_f h(x) = \frac{\partial(L_f h)}{\partial x} g(x) = [0 \ 1] \begin{bmatrix} 0 \\ 1 \end{bmatrix} = 1,$$

and thus we see that the Vand der Pol oscillator system has relative degree 2 at any point x_o .

However, if the output function is, for instance $y = h(x) = \sin x_2$, then $L_g h(x) = \cos x_2$. The system has relative degree 1 at any point x_o , provided that $(x_o)_2 \neq (2k+1)\pi/2$. If the point x_o is such that this condition is violated, no relative degree can be defined.

As another example, consider a *linear system* in the state-space form

$$\dot{x} = Ax + Bu, \quad y = Cx.$$

In this case, since $f(x) = Ax$, $g(x) = B$, $h(x) = Cx$, it can be seen that

$$\begin{aligned} L_f^k h(x) &= CA^k x, & \text{and therefore,} \\ L_g L_f^k h(x) &= CA^k B. \end{aligned}$$

Thus, the integer r is characterized by the conditions

$$\begin{aligned} CA^k B &= 0, & \text{for all } k < r-1 \\ CA^{r-1} B &\neq 0, & \text{otherwise.} \end{aligned}$$

It is well-known that the integer satisfying these conditions is exactly equal to the *difference* between the degree of the denominator polynomial and the degree of the numerator polynomial of the transfer function

$$H(s) = C(sI - A)^{-1}B$$

of the system.

Approximative Feedback Linearization

Consider a SISO system

$$\dot{x} = f(x) + g(x)u, \tag{2.93}$$

where f and g are smooth vector-fields defined on a compact contractible region M of $\mathbb{R}^n$ containing the origin. (Typically, M is a closed ball in $\mathbb{R}^n$.) We assume that $f(0) = 0$, i.e., that the origin is an equilibrium for $\dot{x} = f(x)$. The classical problem of feedback linearization can be stated as follows: find in a neighborhood of the origin a smooth change of coordinates $z = \Phi(x)$ (a local diffeomorphism) and a smooth feedback law $u = k(x) + l(x)u_{new}$ such that the closed-loop system in the new coordinates with new control is linear,

$$\dot{z} = Az + Bu_{new},$$

and controllable (see [BH96]). We usually require that $\Phi(0) = 0$. We assume that the system (4.1) has the *linear controllability* property

$$\dim(\text{span}\{g, Ad_f g, \dots, Ad_f^{n-1} g\}) = n, \quad \text{for all } x \in M \tag{2.94}$$

(where Ad_f^i are iterated Lie brackets of f and g). We define the *characteristic distribution* for (4.1)

$$\mathcal{D} = \text{span}\{g, Ad_f g, \dots, Ad_f^{m-2} g\},$$

which is an $(n-1)$ D smooth distribution by assumption of linear controllability (2.94). We call any nowhere vanishing 1-form ω annihilating $\mathcal{D}$ a characteristic 1-form for (4.1). All the characteristic 1-forms for (4.1) can be represented as multiples of some fixed characteristic 1-form ω_0 by a smooth nowhere vanishing function (zero-form) β . Suppose that there is a nonvanishing β so that $\beta\omega_0$ is exact, i.e., $\beta\omega_0 = d\alpha$ for some smooth function α , where d denotes the exterior derivative. Then ω_0 is called *integrable* and is called an integrating factor for ω_0 . The following result is standard in nonlinear control: Suppose that the system (4.1) has the linear controllability property (2.94) on M . Let $\mathcal{D}$ be the characteristic distribution and ω_0 be a characteristic 1-form for (4.1). The following statements are equivalent:

1. Equation (4.1) is feedback linearizable in a neighborhood of the origin in M ;
2. $\mathcal{D}$ is involutive in a neighborhood of the origin in M ; and
3. ω_0 is integrable in a neighborhood of the origin in M .

As is well known, a generic nonlinear system is not feedback linearizable for $n > 2$. However, in some cases, it may make sense to consider *approximate feedback linearization*.

Namely, if one can find a feedback linearizable system close to (4.1), there is hope that a control designed for the feedback linearizable system and applied to (4.1) will give satisfactory performance if the feedback linearizable system is close enough to (4.1). The first attempt in this direction goes back to [Kre84], where it was proposed to apply to (4.1) a change of variables and feedback that yield a system of the form

$$\dot{z} = Az + B u_{new} + O(z, u_{new}),$$

where the term $O(z, u_{new})$ contains higher-order terms. The aim was to make $O(z, u_{new})$ of as high order as possible. Then we can say that the system (4.1) is approximately feedback linearized in a small neighborhood of the origin. Later [HT93] introduced a new algorithm to achieve the same goal with fewer steps.

Another idea has been investigated in [HSK92]. Roughly speaking, the idea was to neglect nonlinearities in (4.1) responsible for the failure of the involutivity condition in above theorem. This approach happened to be successful in the ball-and-beam system, when neglect of centrifugal force acting on ball yielded a feedback linearizable system. Application of a control scheme designed for the system with centrifugal force neglected to the original system gave much better results than applying a control scheme based on classical Jacobian linearization. This approach has been further investigated

in [XH94, XH95] for the purpose of approximate feedback linearization about the manifold of constant operating points. However, a general approach to deciding which nonlinearities should be neglected to get the best approximation has not been set forth.

All of the above-mentioned work dealt with applying a change of coordinates and a preliminary feedback so that the resulting system looks like linearizable part plus nonlinear terms of highest possible order around an equilibrium point or an equilibrium manifold. However, in many applications one requires a large region of operation for the nonlinearizable system. In such a case, demanding the nonlinear terms to be neglected to be of highest possible order may, in fact, be quite undesirable. One might prefer that the nonlinear terms to be neglected be small in a uniform sense over the region of operation. In this section we present an approach to approximate feedback linearization that uses a change of coordinates and a preliminary feedback to put a system (4.1) in a perturbed Brunovsky form,

$$\dot{z} = Az + B u_{new} + P(z) + Q(z) u_{new}, \quad (2.95)$$

where $P(z)$ and $Q(z)$ vanish at $z = 0$ and are ‘small’ on M . We get upper bounds on uniform norms of P and Q (depending on some measures of noninvolutivity of $\mathcal{D}$) on any compact, contractible M .

A different, indirect approach was presented in [BH96]. In this section, the authors present an approach for finding feedback linearizable systems that approximate a given SISO nonlinear system on a given compact region of the state-space. First, they it is shown that if the system is close to being involutive, then it is also close to being linearizable. Rather than working directly with the characteristic distribution of the system, the authors work with characteristic 1-forms, i.e., with the 1-forms annihilating the characteristic distribution. It is shown that homotopy operators can be used to decompose a given characteristic 1-form into an exact and an antiexact part. The exact part is used to define a change of coordinates to a normal form that looks like a linearizable part plus nonlinear perturbation terms. The nonlinear terms in this normal form depend continuously on the antiexact part, and they vanish whenever the antiexact part does. Thus, the antiexact part of a given characteristic 1-form is a measure of nonlinearizability of the system. If the nonlinear terms are small, by neglecting them we get a linearizable system approximating the original system. One can design control for the original system by designing it for the approximating linearizable system and applying it to the original one. We apply this approach for design of locally stabilizing feedback laws for nonlinear systems that are close to being linearizable.

Let us start with approximating characteristic 1-forms by exact forms using *homotopy operators*. Namely, on any contractible region M one can define a linear operator H that satisfies

$$\omega = d(H\omega) + Hd\omega \quad (2.96)$$

for any form ω . The homotopy identity (2.96) allows to decompose any given 1-form into the *exact part* $d(H\omega)$ and an ‘error part’ $\epsilon = Hd\omega$, which we call the *antieract part of* ω . For given ω_0 annihilating $\mathcal{D}$ and a scaling factor β we define $\alpha_\beta = H\beta\omega_0$ and $\epsilon_\beta = Hd\beta\omega_0$. The 1-form ϵ_β measures how exact $\omega_\beta = \beta\omega_0$ is. If it is zero, then ω_β is exact and the system (4.1) is linearizable, and the zero-form α_β and its first $n - 1$ Lie derivatives along f are the new coordinates. In the case that ω_0 is not exactly integrable, i.e., when no exact integrating factor β exists, we choose β so that $d\beta\omega_0$ is *smallest* in some sense (because this also makes ϵ_β small). We call this β an *approximate integrating factor for* ω_0 . We use the zero-form α_β and its first $n - 1$ Lie derivatives along f as the new coordinates as in the linearizable case. In those new coordinates the system (4.1) is in the form

$$\dot{z} = Az + Bru + Bp + Eu,$$

where r and p are smooth functions, $r \neq 0$ around the origin, and the term E (the obstruction to linearizability) depends linearly on ϵ_β and some of its derivatives. We choose $u = r^{-1}(u_{new} - p)$, where u_{new} is a new control variable. After this change of coordinates and control variable the system is of the form (2.95) with $Q = r^{-1}E$, $P = -r^{-1}pE$. We get estimates on the uniform norm of Q and P (via estimates on r , p , and E) in terms of the error 1-form ϵ_β , for any fixed β , on any compact, contractible manifold M . Most important is that Q and P depend in a continuous way on ϵ_β and some of its derivatives, and they vanish whenever ϵ does (see [BH96]).

Controllability

Linear Controllability

Recall that a system is said to be *controllable* if the set of all states it can reach from initial state $x_0 = x(0)$ at the fixed time $t = T$ contains a ball $\mathcal{B}$ around x_0 . Again, a system is called *small time locally controllable* (STLC) iff the ball $\mathcal{B}$ for $t \leq T$ contains a neighborhood of x_0 .¹³

In the case of a linear system in the standard state-space form

$$\dot{x} = Ax + Bu, \tag{2.97}$$

where A is the $n \times n$ *state matrix* and B is the $m \times n$ *input matrix*, all controllability definitions coincide, i.e.,

$$0 \rightarrow x(T), \quad x(0) \rightarrow 0, \quad x(0) \rightarrow x(T),$$

where T is either fixed or free.

¹³ The above definition of controllability tells us only whether or not something can reach an open neighborhood of its starting point, but does not tell us how to do it. That is the point of the *trajectory generation*.

Rank condition states: System (2.97) is controllable iff the matrix

$$W_n = (B \ AB \ \dots \ A^{n-1}B) \quad \text{has full rank.}$$

In the case of nonlinear systems the corresponding result is get using the formalism of Lie brackets, as Lie algebra is to nonlinear systems as matrix algebra is to linear systems.

Nonlinear Controllability

Nonlinear MIMO-systems are generally described by differential equations of the form (see [Isi89, NS90, Goo98]):

$$\dot{x} = f(x) + g_i(x) u^i, \quad (i = 1, \dots, n), \quad (2.98)$$

defined on a smooth n -manifold M , where $x \in M$ represents the state of the control system, $f(x)$ and $g_i(x)$ are vector-fields on M and the u^i are control inputs, which belong to a set of *admissible controls*, $u^i \in U$. The system (2.98) is called *driftless*, or *kinematic*, or *control linear* if $f(x)$ is identically zero; otherwise, it is called a *system with drift*, and the vector-field $f(x)$ is called the *drift term*. The flow $\phi_t^g(x_0)$ represents the solution of the differential equation $\dot{x} = g(x)$ at time t starting from x_0 . Geometrical way to understand the *controllability* of the system (2.98) is to understand the geometry of the vector-fields $f(x)$ and $g_i(x)$.

Example: Car-Parking Using Lie Brackets. In this popular example, the driver has two different transformations at his disposal. He/she can turn the steering wheel, or he/she can drive the car forward or back. Here, we specify the state of a car by four coordinates: the (x, y) coordinates of the center of the rear axle, the direction θ of the car, and the angle ϕ between the front wheels and the direction of the car. L is the constant length of the car. Therefore, the configuration manifold of the car is 4D, $M = (x, y, \theta, \phi)$.

Using (2.98), the driftless car kinematics can be defined as:

$$\dot{x} = g_1(x) u_1 + g_2(x) u_2, \quad (2.99)$$

with two vector-fields $g_1, g_2 \in \mathcal{X}^k(M)$.

The infinitesimal transformations will be the vector-fields

$$g_1(x) \equiv \text{DRIVE} = \cos \theta \frac{\partial}{\partial x} + \sin \theta \frac{\partial}{\partial y} + \frac{\tan \phi}{L} \frac{\partial}{\partial \theta} \equiv \begin{pmatrix} \cos \theta \\ \sin \theta \\ \frac{1}{L} \tan \phi \\ 0 \end{pmatrix},$$

and

$$g_2(x) \equiv \text{STEER} = \frac{\partial}{\partial \phi} \equiv \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}.$$

Now, STEER and DRIVE do not commute; otherwise we could do all your steering at home before driving of on a trip. Therefore, we have a Lie bracket

$$[g_2, g_1] \equiv [\text{STEER}, \text{DRIVE}] = \frac{1}{L \cos^2 \phi} \frac{\partial}{\partial \theta} \equiv \text{ROTATE}.$$

The operation $[g_2, g_1] \equiv \text{ROTATE} \equiv [\text{STEER}, \text{DRIVE}]$ is the infinitesimal version of the sequence of transformations: steer, drive, steer back, and drive back, i.e.,

$$\{\text{STEER}, \text{DRIVE}, \text{STEER}^{-1}, \text{DRIVE}^{-1}\}.$$

Now, ROTATE can get us out of some parking spaces, but not tight ones: we may not have enough room to ROTATE out. The usual tight parking space restricts the DRIVE transformation, but not STEER. A truly tight parking space restricts STEER as well by putting your front wheels against the curb.

Fortunately, there is still another commutator available:

$$\begin{aligned} [g_1, [g_2, g_1]] &\equiv [\text{DRIVE}, [\text{STEER}, \text{DRIVE}]] = [[g_1, g_2], g_1] \equiv \\ [\text{DRIVE}, \text{ROTATE}] &= \frac{1}{L \cos^2 \phi} \left(\sin \theta \frac{\partial}{\partial x} - \cos \theta \frac{\partial}{\partial y} \right) \equiv \text{SLIDE}. \end{aligned}$$

The operation $[[g_1, g_2], g_1] \equiv \text{SLIDE} \equiv [\text{DRIVE}, \text{ROTATE}]$ is a displacement at right angles to the car, and can get us out of any parking place. We just need to remember to steer, drive, steer back, drive some more, steer, drive back, steer back, and drive back:

$$\{\text{STEER}, \text{DRIVE}, \text{STEER}^{-1}, \text{DRIVE}, \text{STEER}, \text{DRIVE}^{-1}, \text{STEER}^{-1}, \text{DRIVE}^{-1}\}.$$

We have to reverse steer in the middle of the parking place. This is not intuitive, and no doubt is part of the problem with parallel parking.

Thus from only two controls u_1 and u_2 we can form the vector-fields $\text{DRIVE} \equiv g_1$, $\text{STEER} \equiv g_2$, $\text{ROTATE} \equiv [g_2, g_1]$, and $\text{SLIDE} \equiv [[g_1, g_2], g_1]$, allowing us to move anywhere in the configuration manifold M . The car kinematics $\dot{x} = g_1 u_1 + g_2 u_2$ is thus expanded as:

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\phi} \end{pmatrix} = \text{DRIVE} \cdot u_1 + \text{STEER} \cdot u_2 \equiv \begin{pmatrix} \cos \theta \\ \sin \theta \\ \frac{1}{L} \tan \phi \\ 0 \end{pmatrix} \cdot u_1 + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \cdot u_2.$$

The *parking theorem* says: One can get out of any parking lot that is larger than the car.

Controllability Condition

Nonlinear controllability is an extension of linear controllability. The nonlinear MIMO system

$$\dot{x} = f(x) + g(x)u \quad \text{is controllable}$$

if the set of vector-fields $\{g, [f, g], \dots, [f^{n-1}, g]\}$ is independent.

For example, for the kinematic car system of the form (2.99), the *nonlinear controllability criterion* reads: If the Lie bracket tree:

$$g_1, g_2, [g_1, g_2], [[g_1, g_2], g_1], [[g_1, g_2], g_2], [[[g_1, g_2], g_1], g_1], [[[g_1, g_2], g_1], g_2],$$

$[[[g_1, g_2], g_2], g_1], [[g_1, g_2], g_2], g_2], \dots$
 – has *full rank* then the system is *controllable* [Isi89, NS90, Goo98]. In this case the combined input

$$(u_1, u_2) = \begin{cases} (1, 0), & t \in [0, \varepsilon] \\ (0, 1), & t \in [\varepsilon, 2\varepsilon] \\ (-1, 0), & t \in [2\varepsilon, 3\varepsilon] \\ (0, -1), & t \in [3\varepsilon, 4\varepsilon] \end{cases}$$

gives the motion $x(4\varepsilon) = x(0) + \varepsilon^2 [g_1, g_2] + O(\varepsilon^3)$, with the flow given by

$$F_t^{[g_1, g_2]} = \lim_{n \rightarrow \infty} \left(F_{\sqrt{t/n}}^{-g_2} F_{\sqrt{t/n}}^{-g_1} F_{\sqrt{t/n}}^{g_2} F_{\sqrt{t/n}}^{g_1} \right)^n.$$

Distributions

In control theory, the set of all possible directions in which the system can move, or the set of all points the system can reach, is of obvious fundamental importance. Geometrically, this is related to *distributions*.

A *distribution* $\Delta \subset \mathcal{X}^k(M)$ on a smooth n D manifold M is a subbundle of its tangent bundle TM , which assigns a subspace of the tangent space $T_x M$ to each point $x \in M$ in a smooth way. The dimension of $\Delta(x)$ over $\mathbb{R}$ at a point $x \in M$ is called the rank of Δ at x .

A distribution Δ is involutive if, for any two vector-fields $X, Y \in \Delta$, their Lie bracket $[X, Y] \in \Delta$.

A function $f \in C^\infty(M)$ is called an *integral* of Δ if $df(x) \in \Delta^0(x)$ for each $x \in M$. An *integral manifold* of Δ is a submanifold N of M such that $T_x N \subset \Delta(x)$ for each $x \in N$. A distribution Δ is *integrable* if, for any $x \in M$, there is a submanifold $N \subset M$, whose dimension is the same as the rank of Δ at x , containing x such that the tangent bundle, TN , is exactly Δ restricted to N , i.e., $TN = \Delta|_N$. Such a submanifold is called the *maximal integral manifold* through x .

It is natural to consider distributions generated by the vector-fields appearing in the above sequence of flows. Consider the distribution defined by

$$\Delta = \text{span}\{f; g_1 \dots g_m\},$$

where the span is taken over the set of smooth real-valued functions. Denote by $\bar{\Delta}$ the *involutive closure* of the distribution Δ , which is the closure of Δ under bracketing. Then, $\bar{\Delta}$ is the smallest subalgebra of $\mathcal{X}^k(M)$ which contains $\{f; g_1 \dots g_m\}$. We will often need to ‘add’ distributions. Since distributions are, pointwise, vector spaces, define the sum of two distributions,

$$(\Delta_1 + \Delta_2)(x) = \Delta_1(x) + \Delta_2(x).$$

Similarly, define the intersection

$$(\Delta_1 \cap \Delta_2)(x) = \Delta_1(x) \cap \Delta_2(x).$$

More generally, we can arrive at a distribution via a *family of vector-fields*, which is a subset $\mathcal{V} \subset \mathcal{X}^k(M)$. Given a family of vector-fields $\mathcal{V}$, we may define a distribution on M by

$$\Delta_{\mathcal{V}}(x) = \langle X(x) | X \in \mathcal{V} \rangle_{\mathbb{R}}.$$

Since $\mathcal{X}^k(M)$ is a Lie algebra, we may ask for the smallest Lie subalgebra of $\mathcal{X}^k(M)$ which contains a family of vector-fields $\mathcal{V}$. It will be denoted as $\overline{\text{Lie}}(\mathcal{V})$, and will be represented by the set of vector-fields on M generated by repeated Lie brackets of elements in $\mathcal{V}$. Let $\mathcal{V}^{(0)} = \mathcal{V}$ and then iteratively define a sequence of families of vector-fields by

$$\mathcal{V}^{(i+1)} = \mathcal{V}^{(i)} \cup \{[X, Y] | X \in \mathcal{V}^{(0)} = \mathcal{V} \text{ and } Y \in \mathcal{V}^{(i)}\}.$$

Now, every element of $\overline{\text{Lie}}(\mathcal{V})$ is a linear combination of repeated Lie brackets of the form

$$[Z_k, [Z_{k-1}, [\dots, [Z_2, Z_1] \dots]]]$$

where $Z_i \in \mathcal{V}$ for $i = 1, \dots, k$.

Foliations

Recall that related to integrable distributions are *foliations*.

The *Frobenius theorem* asserts that integrability and involutivity are equivalent, at least locally. Thus, associated with an involutive distribution is a partition Φ of M into disjoint connected immersed submanifolds called *leaves*. This partition Φ is called a *foliation*. More precisely, a foliation $\mathcal{F}$ of a smooth manifold M is a collection of disjoint immersed submanifolds of M whose disjoint union equals M . Each connected submanifold of $\mathcal{F}$ is called a *leaf* of the foliation. Given an integrable distribution Δ , the collection of maximal integral manifolds for Δ defines a foliation on M , denoted by $\mathcal{F}_D$.

A foliation $\mathcal{F}$ of M defines an equivalence relation on M whereby two points in M are equivalent if they lie in the same leaf of $\mathcal{F}$. The set of equivalence classes is denoted $M/\mathcal{F}$ and is called the *leaf space* of $\mathcal{F}$. A foliation $\mathcal{F}$ is said to be simple if $M/\mathcal{F}$ inherits a manifold structure so that the projection from M to $M/\mathcal{F}$ is a surjective submersion.

In control theory, foliation leaves are related to the set of points that a control system can reach starting from a given initial condition. A foliation Φ of M defines an equivalence relation on M whereby two points in M are equivalent if they lie in the same leaf of Φ . The set of equivalence classes is denoted M/Φ and is called the *leaf space* of Φ .

Philip Hall Basis

Given a set of vector-fields $\{g_1 \dots g_m\}$, define the *length* of a *Lie product* as

$$l(g_i) = 1, \quad l([A, B]) = l(A) + l(B), \quad (\text{for } i = 1, \dots, m),$$

where A and B may be Lie products. A *Philip Hall basis* is an ordered set of Lie products $H = \{B_i\}$ satisfying:

1. $g_i \in H$, $(i = 1, \dots, m)$;
2. If $l(B_i) < l(B_j)$, then $B_i < B_j$; and
3. $[B_i, B_j] \in H$ iff
 - (a) $B_i, B_j \in H$ and $B_i < B_j$, and
 - (b) either $B_j = g_k$ for some k or $B_j = [B_l, B_r]$ with $B_l, B_r \in H$ and $B_l \leq B_i$.

Essentially, the ordering aspect of the Philip Hall basis vectors accounts for skew symmetry and Jacobi identity to determine a basis.

Abstract Control System

In general, a nonlinear control system Σ can be represented as a triple (Σ, M, f) , where M is the system's *state-space* manifold with the tangent bundle TM and the general fibre bundle E , and f is a smooth map, such that the following bundle diagram commutes [Man98]

$$\begin{array}{ccc} E & \xrightarrow{\psi} & TM \\ & \searrow \pi & \swarrow \pi_M \\ & M & \end{array}$$

where $\psi : (x, u) \mapsto (x, f(x, u))$, π_M is the natural projection of TM on M , the projection $\pi : E \rightarrow M$ is a smooth fibre bundle, and the fibers of E represent the *input spaces*. If one chooses fibre-respecting coordinates (x, u) for E , then locally this definition reduces to $\psi : (x, u) \mapsto (x, \psi(x, u))$, i.e.,

$$\dot{x} = \psi(x, u).$$

The specific form of the map ψ , usually used in nonlinear control, is $\psi : (x, u) \mapsto (x, f(x) + g(x, u))$, with $g(x, 0) = 0$, producing the standard nonlinear system equation

$$\dot{x} = f(x) + g(x, u).$$

Adaptive Lie-Derivative Control

In this subsection we develop the concept of *machine learning* in the framework of Lie-derivative control formalism (see (2.2.2) above). Consider an n D, SISO system in the standard affine form (2.90), rewritten here for convenience:

$$\dot{x}(t) = f(x) + g(x)u(t), \quad y(t) = h(x), \quad (2.100)$$

As already stated, the feedback control law for the system (2.100) can be defined using Lie derivatives $\mathcal{L}_f h$ and $\mathcal{L}_g h$ of the system's output h along the vector-fields f and g .

If the SISO system (2.100) is a relatively simple (quasilinear) system with *relative degree*¹⁴ = 1 it can be rewritten in a quasilinear form

$$\dot{x}(t) = \gamma_i(t) f_i(x) + d_j(t) g_j(x) u(t), \quad (2.101)$$

where γ_i ($i = 1, \dots, n$) and d_j ($j = 1, \dots, m$) are system's parameters, while f_i and g_j are smooth vector-fields.

In this case the feedback control law for *tracking* the *reference signal* $y_R = y_R(t)$ is defined as (see [Isi89, NS90])

$$u = \frac{-\mathcal{L}_f h + \dot{y}_R + \alpha(y_R - y)}{\mathcal{L}_g h}, \quad (2.102)$$

where α denotes the feedback gain.

Obviously, the problem of reference signal tracking is relatively simple and straightforward if we know all the system's parameters $\gamma_i(t)$ and $d_j(t)$ of (2.101). The question is can we apply a similar control law if the system parameters are unknown?

Now we have much harder problem of *adaptive signal tracking*. However, it appears that the feedback control law can be actually cast in a similar form (see [SI89],[Gom94]):

$$\hat{u} = \frac{-\widehat{\mathcal{L}_f h} + \dot{y}_R + \alpha(y_R - y)}{\widehat{\mathcal{L}_g h}}, \quad (2.103)$$

where Lie derivatives $\mathcal{L}_f h$ and $\mathcal{L}_g h$ of (2.102) have been replaced by their estimates $\widehat{\mathcal{L}_f h}$ and $\widehat{\mathcal{L}_g h}$, defined respectively as

$$\widehat{\mathcal{L}_f h} = \hat{\gamma}_i(t) \mathcal{L}_{f_i} h, \quad \widehat{\mathcal{L}_g h} = \hat{d}_j(t) \mathcal{L}_{g_j} h,$$

¹⁴ Relative degree equals the number of differentiations of the output function y required to have the input u appear explicitly. Technically, the system (2.100) is said to have relative degree r at the point x^0 if (see [Isi89, NS90])

- (i) $\mathcal{L}_g \mathcal{L}_f^k h(x) = 0$ for all x in a neighborhood of x^0 and all $k < r - 1$, and
- (ii) $\mathcal{L}_g \mathcal{L}_f^{r-1} h(x^0) \neq 0$,

where $\mathcal{L}_f^k h$ denotes the k th Lie derivative of h along f .

in which $\widehat{\gamma}_i(t)$ and $\widehat{d}_j(t)$ are the estimates for $\gamma_i(t)$ and $d_j(t)$.

Therefore, we have the straightforward control law even in the uncertain case, provided that we are able to estimate the unknown system parameters. Probably the best known *parameter update law* is based on the so-called *Lyapunov criterion* (see [SI89]) and given by

$$\dot{\psi} = -\gamma \epsilon W, \quad (2.104)$$

where $\psi = \{\gamma_i - \widehat{\gamma}_i, d_j - \widehat{d}_j\}$ is the parameter estimation error, $\epsilon = y - y_R$ is the output error, and γ is a positive constant, while the matrix W is defined as:

$$W = [W_1^T W_2^T]^T, \quad \text{with}$$

$$W_1 = \begin{bmatrix} \mathcal{L}_{f_1} h \\ \vdots \\ \mathcal{L}_{f_n} h \end{bmatrix}, \quad W_2 = \begin{bmatrix} \mathcal{L}_{g_1} h \\ \vdots \\ \mathcal{L}_{g_m} h \end{bmatrix} \cdot \frac{-\widehat{\mathcal{L}_f h} + \dot{y}_R + \alpha(y_R - y)}{\widehat{\mathcal{L}_g h}}.$$

The proposed adaptive control formalism (2.103–2.104) can be efficiently applied wherever we have a problem of tracking a given signal with an output of a SISO-system (2.100–2.101) with unknown parameters.

A generalization to MIMO systems would be really interesting. It would give us a rigorous closed-form model for model-free neural networks (see below).

2.2.3 Complex Adaptive Systems

According to [AEH05], a *complex adaptive system* (CAS) consists of inhomogeneous, interacting adaptive agents, where the word *adaptive* means *capable of learning*. An *emergent property* of a CAS is a property of the system as a whole which does not exist at the individual elements (agents) level.

Typical CAS examples are the brain, the immune system, the economy, social systems, ecology, etc... Most of living systems are CAS.

Therefore to understand a complex system one has to study the system as a whole and not to decompose it into its constituents. This totalistic approach is against the standard reductionist one, which tries to decompose any system to its constituents and hopes that by understanding the elements one can understand the whole system.

The standard approaches to CAS modeling are: (i) ODEs, difference equations and PDEs; (ii) Cellular automata (CA) [Ila01]; (iii) Evolutionary game theory [HS98]; (iv) Various agent based models; (v) Complex networks (see previous subsection); and (vi) Fractional calculus [Sta00]. Most of these approaches are included in [Boc04].

Both the ODE and PDE approaches have some difficulties as follows [LSA03]: (i) ODE and PDE assumes that local fluctuations have been smoothed

out, (ii) typically they neglect correlations between movements of different species, and (iii) they assume instantaneous results of interactions.

Most biological systems show delay and do not satisfy the above assumptions. They concluded that a cellular automata (CA) [Ila01] type system called microscopic simulation is more suitable to model complex biological systems. We agree that CA type systems are more suitable to model complex biological systems but such systems suffer from a main drawback namely the difficulty of getting analytical results. The known analytical results about CA type systems are very few compared to the known results about ODE and PDE.

A compromise was presented in [AEH05] in the form of a PDE, which avoids the delay and the correlations drawbacks. It is called *telegraph reaction diffusion equation* (TRD). To overcome the non-delay weakness in the *Fick law*,¹⁵ it is replaced by

$$J(x, t) + \tau \partial_t J(x, t) = -D \partial_x c, \quad (2.105)$$

where the flux $J(x, t)$ relaxes, with some given characteristic time constant τ and c is the concentration of the diffusing substance. Combining (2.105) with the equation of continuity, one gets the *modified diffusion equation*, or the *telegraph equation*, $\dot{c} = (D - \tau) \partial_{x^2} c$. The corresponding TRD equation is given by

$$\tau \ddot{c} + \left(1 - \frac{df(c)}{dc}\right) \dot{c} = D \partial_{x^2} c + f(c), \quad (2.106)$$

where $f(c)$ is a polynomial in c . Moreover it is known that TRD results from correlated random walk. This supports the conclusion that TRD equation (2.106) is more suitable for modeling complex systems than the usual diffusion one.

For example, a *human immune system* as a CAS was elaborated in [SC01]. The emergent properties of the immune system (IS) included:

- * The ability to distinguish any substance (typically called antigen Ag) and determine whether it is damaging or not. If Ag is non-damaging (damaging) then, typically, IS tolerates it (responds to it).
- * If it decides to respond to it then IS determines whether to eradicate it or to contain it.
- * The ability to memorize most previously encountered Ag, which enables it to mount a more effective reaction in any future encounters. This is the basis of vaccination processes.
- * IS is complex thus it has a network structure.

¹⁵ Recall that *diffusion* through a fluid is can be described by the *Fick equation*

$$\partial_t T = D \nabla^2 c, \quad D > 0,$$

where T is the temperature, c is the concentration of a certain substance dissolved in the fluid and D is the mass diffusivity.

- * The immune network is not homogeneous since there are effectors with many connections and others with low number of connections.
- * The Ag, which enters our bodies, has extremely wide diversity. Thus mechanisms have to exist to produce immune effectors with constantly changing random specificity to be able to recognize these Ag. Consequently IS is an adaptive complex system.
- * Having said that, one should notice that the wide diversity of IS contains the danger of autoimmunity (attacking the body). Thus mechanisms that limit autoimmunity should exist.
- * In addition to the primary clonal deletion mechanism, two further brilliant mechanisms exist: The first is that the IS network is a threshold or 'window' one i.e., no activation exists if the Ag quantity is too low or too high (This is called low and high zone tolerance).
- * Thus an auto reactive immune effector (i.e., an immune effector that attacks the body to which it belongs) will face so many self-antigens that it has to be suppressed due to the high zone tolerance mechanism.
- * Another mechanism against autoimmunity is the second signal given by antigen presenting cells (APC). If the immune effector is self reactive then, in most cases, it does not receive the second signal thus it becomes anergic.
- * Also long term memory can be explained by the phenomena of high and low zone tolerance where IS tolerates Ag if its quantity is too high or too low. So persisting Ag is possible and continuous activation of immune effectors may occur.
- * There is another possible explanation for long term memory using the immune system (extremal dynamics).
- * Thus design principles of IS can explain important phenomena of IS.

The following Summary on CAS was given in [AEH05] (see also the literature cited there):

- (i) CAS should be studied as a whole hence reductionist point of view may not be reliable in some cases.
- (ii) CAS are open with nonlinear local interactions hence: (1) Long range prediction is highly unlikely. (2) When studying a CAS take into consideration the effects of its perturbation on related systems e.g. perturbation of lake Victoria has affected mosquitoes' numbers hence the locals quality of life. This is also relevant to the case of natural disasters where an earthquake at a city can cause a widespread power failure at other cities. (3) Expect side effects to any 'seemingly wise' decision. (4) Mathematical and computer models may be helpful in reducing such side effects.
- (iii) Optimization in CAS should be multi-objective (not single objective).
- (iv) CAS are very difficult to control. Interference at highly connected sites may be a useful approach. The interlinked nature of CAS elements complicates both the unpredictability and controllability problems. It also plays an important role in innovations spread.

- (v) Memory effects should not be neglected in CAS. This lends more support for the proposed TRD equation (2.106). Also, memory games have been studied. Also, delay and fractional calculus are relevant to CAS.
- (vi) Mathematical topics motivated by CAS include ODE and PDE (non-autonomous, delayed, periodic coefficients, stability and persistence), multi-objective optimization (including biologically motivated methods e.g., ant colony optimization, extremal optimization, genetic algorithms, etc.), difference equations, cellular automata, networks, fractional calculus, control (e.g., bounded delayed control of distributed systems), game theory, nonlinear dynamics and fuzzy mathematics.

2.2.4 Complex Networks and Synchronization Dynamics

Recall that *networks of dynamical elements* serve as natural models for a variety of systems, with examples ranging from cell biology to epidemiology to the Internet (see, e.g., [Str01]). Many of these complex networks display common structural features, such as the *small-world* [WS98] and *scale-free* properties [BA99]. The so-called *small-world networks* (SWNs, see next section) exhibit short average distance between nodes and high clustering [WS98], while the *scale-free networks* (SFNs) are characterized by an algebraic, highly heterogeneous distribution of degrees (number of links per node) [BA99]. The interplay between structure and dynamics has attracted a great deal of attention, especially in connection with the problem of *synchronization of coupled oscillators* [Wat99, BP02, NML02]. The ability of a network to synchronize is generally enhanced in both SWNs and random SFNs as compared to *regular lattices* [Wat99, BP02, NML02]. This enhancement was previously believed to be due to the decrease of the average distance between oscillators. Recently, it was shown that random networks with strong heterogeneity in the degree distribution, such as SFNs, are much more difficult to synchronize than *random homogeneous networks* [NML02], even though the former display smaller average path length [CH03]. This suggests that, although structurally advantageous [AJB00], the scale-free property may be dynamically detrimental [MZK05].

A basic assumption has been that the oscillators are coupled symmetrically and with the same coupling strength. Under the *assumption of symmetric coupling*, the maximum synchronizability is indeed achieved when the coupling strength is uniform. But to get a better synchronizability, the couplings are not necessarily symmetrical. Many realistic networks are actually directed [Str01] and weighted [New01]. In particular, the communication capacity of a node is likely to saturate when the degree becomes large.

In this subsection, following [MZK05], we study the impact that asymmetry and saturation of connection strength have on the *synchronization dynamics* on complex networks. As a prime example, we consider complete synchronization of linearly coupled identical oscillators defined by

$$\dot{x}_i = f(x_i) - \sigma G_{ij} h(x_j), \quad (i = 1, \dots, N), \quad (2.107)$$

where $f = f(x)$ describes the dynamics of each individual oscillator, $h = h(x)$ is the output function, $G = (G_{ij})$ is the coupling matrix, and σ is the overall coupling strength. In the case of symmetrically coupled oscillators with uniform coupling strength, G is the usual (symmetric) Laplacian matrix¹⁶ $L = (L_{ij})$. For $G_{ij} = L_{ij}$, heterogeneity in the degree distribution suppresses synchronization [NML02]. In order to enhance the synchronizability of heterogeneous networks, we present to scale the coupling strength by the degrees of the nodes. For specificity, we consider

$$G_{ij} = L_{ij}/k_i^\beta, \quad (2.108)$$

where k_i is the degree of node i and β is a tunable parameter. The underlying network associated with the Laplacian matrix L is undirected and unweighted, but with the introduction of the weights in (3.460), the network of couplings becomes not only weighted but also directed because the resulting matrix G is in general asymmetric. We say that the network or coupling is weighted when $\beta \neq 0$ and unweighted when $\beta = 0$.

The variational equations governing the linear stability of a synchronized state $\{x_i(t) = s(t), \text{ for all } i\}$ can be diagonalized into N blocks of the form

$$\dot{\eta} = [Df(s) - \alpha Dh(s)] \eta, \quad \text{where} \quad \alpha = \sigma \lambda_i,$$

and λ_i are the eigenvalues of the coupling matrix G , ordered as $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_N$ (see below). The largest *Lyapunov exponent* $\Lambda(\alpha)$ of this equation can be regarded as a *master stability function*, which determines the linear stability of the synchronized state [PC98]: the synchronized state is stable if $\Lambda(\sigma \lambda_i) < 0$ for $i = 2, \dots, N$.¹⁷ For many widely studied oscillatory systems [BP02, PC98], the master stability function $\Lambda(\alpha)$ is negative in a finite interval (α_1, α_2) . Therefore, the network is synchronizable for some σ when the eigen-ratio $R = \lambda_N/\lambda_2$ satisfies $R < \alpha_2/\alpha_1$. The ratio α_2/α_1 depends only on the dynamics (f , h , and s), while the eigen-ratio R depends only on the coupling matrix G . The problem of synchronization is then reduced to the analysis of eigenvalues of the coupling matrix [BP02]: the smaller the eigen-ratio R the more synchronizable the network.

Here we show that, as a function of β , the eigen-ratio R has a global minimum at $\beta = 1$. In large networks with some degree of randomness, the eigen-ratio at $\beta = 1$ is primarily determined by the average degree k of the network and *does not* depend on the degree distribution and system size, in sharp contrast with the case of unweighted coupling ($\beta = 0$), where synchronization is strongly suppressed as the heterogeneity or number of oscillators is increased [MZK05]. Furthermore, we show that the total cost involved in the network coupling is significantly reduced for $\beta = 1$ when compared to $\beta = 0$.

¹⁶ The diagonal entries are $L_{ii} = k_i$. The off-diagonal entries are $L_{ij} = -1$, if nodes i and j are connected, and 0 otherwise.

¹⁷ Recall that the eigenvalue λ_1 corresponds to a mode parallel to the synchronization manifold.

As a result, structural robustness [AJB00] and improved synchronizability can coexist in scale-free and other heterogeneous networks.

In matrix notation, equation (3.460) can be written as

$$G = D^{-\beta}L, \quad \text{where} \quad D = \text{diag}\{k_1, k_2, \dots, k_N\}$$

is the diagonal matrix of degrees. From the identity

$$\det(D^{-\beta}L - \lambda I) = \det(D^{-\beta/2}LD^{-\beta/2} - \lambda I),$$

valid for any λ , we have that the spectrum of eigenvalues of matrix G is equal to the spectrum of a symmetric matrix defined as

$$H = D^{-\beta/2}LD^{-\beta/2}.$$

As a result, all the eigenvalues of matrix G are real. Moreover, because H is positive semidefinite, all the eigenvalues are nonnegative and, because the rows of G have zero sum, the smallest eigenvalue λ_1 is always zero, as assumed above. If the network is connected, $\lambda_2 > 0$ for any finite β [MZK05].

We first examine the dependence on β . Physically, we expect the synchronizability to be strongly influenced by the strength of the input coupling at each oscillator. When $\beta < 1$, oscillators with larger degree are more strongly coupled than oscillators with smaller degree. When $\beta > 1$, the opposite happens. Because (α_1, α_2) is finite, for the network to synchronize, the overall coupling strength σ must be large enough to synchronize the least coupled oscillators and small enough to synchronize the most coupled ones (i.e., the synchronizability of these oscillators is expected to be primarily determined by the modes associated with the eigenvalues λ_2 and λ_N , respectively). Therefore, for both $\beta < 1$ and $\beta > 1$, some oscillators are more strongly coupled than others, and the ability of the network to synchronize is limited by those oscillators that are least and most strongly coupled. We then expect the network to achieve maximum synchronizability at $\beta = 1$. In [MZK05], the numerical verification of this hypothesis was shown on three different models of SFNs, defined as follows:

- (A) *Random SFNs* [NSW01] — Each node is assigned to have a number k_i of ‘half-links’ according to the probability distribution $P(k) \sim k^{-\gamma}$, where γ is a scaling exponent and $k \geq k_{min}$. The network is generated by randomly connecting these half-links to form links, prohibiting self- and repeated links. In the limit $\gamma = \infty$, all the nodes have the same degree $k = k_{min}$.
- (B) *Networks with expected scale-free sequence* [CLV03] — The network is generated from a sequence $\tilde{k}_1, \tilde{k}_2, \dots, \tilde{k}_N$, where $\max_i \tilde{k}_i^2 < \sum_i \tilde{k}_i$, so that links are independently assigned to each pair of nodes (i, j) with probability $p_{ij} = \tilde{k}_i \tilde{k}_j / \sum_i \tilde{k}_i$. When the expected degrees $\tilde{k}_i \geq \tilde{k}_{min}$ follow the distribution $P(\tilde{k}) \sim \tilde{k}^{-\gamma}$, we have a network with expected scale-free sequence.

(C) *Growing SFNs* [LLY02] — We start with a fully connected network with m nodes and at each time step a new node with m links is added to the network. Each new link is connected to a node i in the network with probability $\Pi_i \sim (1-p)k_i + p$, where $0 \leq p \leq 1$ is a tunable parameter. For large degrees, the scaling exponent of the resulting network is $\gamma = 3 + p[m(1-p)]^{-1}$. For $p = 0$, we recover the Barabási–Albert model [BA99].

A pronounced minimum for the eigen-ratio R at $\beta = 1$ is observed in each case. A similar minimum for R at $\beta = 1$ is also observed in many other models of complex networks, including the Watts–Strogatz model [WS98] of SWNs. The only exception is the class of homogeneous networks, where all the nodes have the same degree k . In this case, the weights can be factored out and R is independent of β .

In heterogeneous networks, the synchronizability is significantly enhanced when the coupling is suitably weighted for SFNs with $\beta = 1$. In SFNs, the heterogeneity (variance) of the degree distribution increases as the scaling exponent γ is reduced. When the coupling is unweighted ($\beta = 0$), the eigen-ratio R increases with heterogeneity, but the eigen-ratio does not increase and may even decrease with heterogeneity when the coupling is weighted ($\beta = 1$). The enhancement is particularly large for small γ , where the networks are highly heterogeneous. The networks become more homogeneous as γ is increased. In the limit $\gamma = \infty$, random SFNs converge to random homogeneous networks with the same degree k_{min} for all nodes, while networks with expected scale-free sequence converge to *Erdős–Rényi random networks* [Bol01], which have links assigned with the same probability between each pair of nodes. The synchronizability is strongly enhanced even in the relatively homogeneous Erdős–Rényi model; such an enhancement occurs also in growing networks. Surprisingly, for $\beta = 1$, the eigen-ratio R turns out to be well approximated by the corresponding eigen-ratio of random homogeneous networks with the same average degree. Moreover, the eigen-ratio R appears to be independent of the system size for $\beta = 1$ in large SFNs, in contrast to the un-weighted case, where R increases strongly with the number of oscillators (see [MZK05] for details).

We now present an approximation for the eigen-ratio R that supports and extends our numerical observations. In what follows we focus on the case $\beta = 1$. Based on results of [CLV03] for random networks with arbitrary expected degrees, which includes important SFNs, we get [MZK05]

$$\max\{1 - \lambda_2, \lambda_N - 1\} = [1 + o(1)] \frac{2}{\sqrt{\tilde{k}}}, \quad (2.109)$$

where $\tilde{k}$ is the average expected degree. This result is rigorous for networks with a given expected degree sequence $\tilde{k}_1, \tilde{k}_2, \dots, \tilde{k}_N$, as defined in the model (B) above. The assumption for this result is $\tilde{k}_{min} \equiv \min_i \tilde{k}_i$ to be large as compared to $\sqrt{\tilde{k}} \ln^3 N$, but our numerical simulations suggest that this as-

sumption can be released considerably because (2.109) is observed to hold for $\tilde{k}_{min}$ as small as $2\sqrt{\tilde{k}}$. Having released this assumption, from (2.109) we have the following explicit upper bound for the eigen-ratio in large networks:

$$R \leq \frac{1 + 2/\sqrt{\tilde{k}}}{1 - 2/\sqrt{\tilde{k}}}. \quad (2.110)$$

Therefore, the eigen-ratio is bounded by a function of the average degree, which does not depend on the system size. Moreover, we expect R to approach the upper bound in (2.110) because the semicircle law holds and the spectrum is symmetric around 1 for $\tilde{k}_{min} \gg \sqrt{\tilde{k}}$ in the thermodynamical limit [CLV03, DM03]. This prediction is confirmed numerically under much weaker conditions, where one can see a remarkable agreement between the approximate and exact values of R for all three models of SFNs. Since the bound in (2.110) does not depend on the degree distribution, this result also explains the agreement between the eigen-ratio for weighted SFNs and the eigen-ratio for random homogeneous networks with the same average degree. A similar agreement is observed in many other complex networks [MZK05].

It is also important to study the influence of degree correlation and clustering on the synchronizability of the networks. The eigen-ratio R generally increases with increasing clustering and assortativity in correlated networks. However, a pronounced global minimum for R as a function of β is always observed at $\beta = 1$. In addition, weighted networks at $\beta = 1$ are much more insensitive to the effects of correlation than their unweighted counterparts. The same tendency is observed in the growing model with aging of [DM03], which has nontrivial clustering and correlation. All together, these suggest that our results are quite robust and expected to hold on real-world networks as well.

Finally, we address the important problem of the *cost* involved in the connections of the network. The cost C is naturally defined as the total strength of all the directed links, i.e.,

$$C = \sigma_{min} \sum_i k_i^{1-\beta}, \quad \text{where} \quad \sigma_{min} = \alpha_1/\lambda_2$$

is the minimum overall coupling strength for the network to synchronize. Strikingly, in heterogeneous networks, the cost for $\beta = 1$ is considerably smaller than the cost for $\beta = 0$. The cost for $\beta = 1$ is well approximated by the cost for random homogeneous networks with the same average degree k . In this case, from above we have $C/(N\alpha_1) = 1/\lambda_2 \approx 1/(1 - 2/\sqrt{k})$, and the cost is reduced as k is increased, approaching $C/(N\alpha_1) = 1$ for large globally coupled networks. Therefore, cost reduction is another important advantage of the weighted coupling [MZK05].

2.2.5 Small-Worlds and Social Networks

Recall that the so-called *small-world networks* have received much attention from researchers in various disciplines, since they were introduced by Watts and Strogatz [WS98] as models of real networks that lie somewhere between being random and being regular.

Watts and Strogatz introduced a simple model for tuning collections of coupled dynamical systems between the two extremes of random and regular networks. In this model, connections between nodes in a regular array are randomly rewired with a probability p , such that $p = 0$ means the network is regularly connected, while $p = 1$ results in a random connection of nodes. For a range of intermediate values of p between these two extremes, the network retains a property of regular networks (a large clustering coefficient) and also acquires a property of random networks (a short characteristic path length between nodes).

Many examples of such small worlds, both natural and human-made, have been discussed [Str01, New00]. For example, a model of a *social network* is given in Figure 2.24, where (a) denotes people (dots) belonging to groups (ellipses), which in turn belong to groups of groups and so on. The largest group corresponds to the entire community. As we go down in this hierarchical organization, each group represents a set of people with increasing social affinity. In the example, there are $l = 3$ hierarchical levels, each representing a subdivision in $b = 3$ smaller groups, and the lowest groups are composed of $g = 11$ people, on average. This defines a *social hierarchy*. The distance between the highlighted individuals i and j in this hierarchy is 3. (b) Each hierarchy can be represented as a tree-like structure. Different hierarchies are correlated, in the sense that distances that are short along one of them are more likely to be short along the others as well. The figure shows an example with $H = 2$ hierarchies, where highlighted in the second hierarchy are those people belonging to group A in the first one. (c) Pairs of people at shorter social distances are more likely to be linked by social ties, which can represent either friendship or acquaintanceship ties (we do not distinguish them here because the ones that are relevant for the problem in question may depend on the social context). The figure shows, for a person in the network, the distribution of acquaintances at social distance $D = 1, 2$, and 3 , where D is the minimum over the distances along all the hierarchies.

Not surprisingly, there has been much interest in the synchronization of dynamical systems connected in a *small-world geometry* [BP02, NML03]. Generically, such studies have shown that the presence of small-world connections make it easier for a network to synchronize, an effect generally attributed to the reduced path length between the linked systems. This has also been found to be true for the special case in which the dynamics of each oscillator is described by a *Kuramoto model* [HCK02a, HCK02b].

Small-world networks are characterized by two numbers: the average path length L and the clustering coefficient C . L , which measures efficiency of

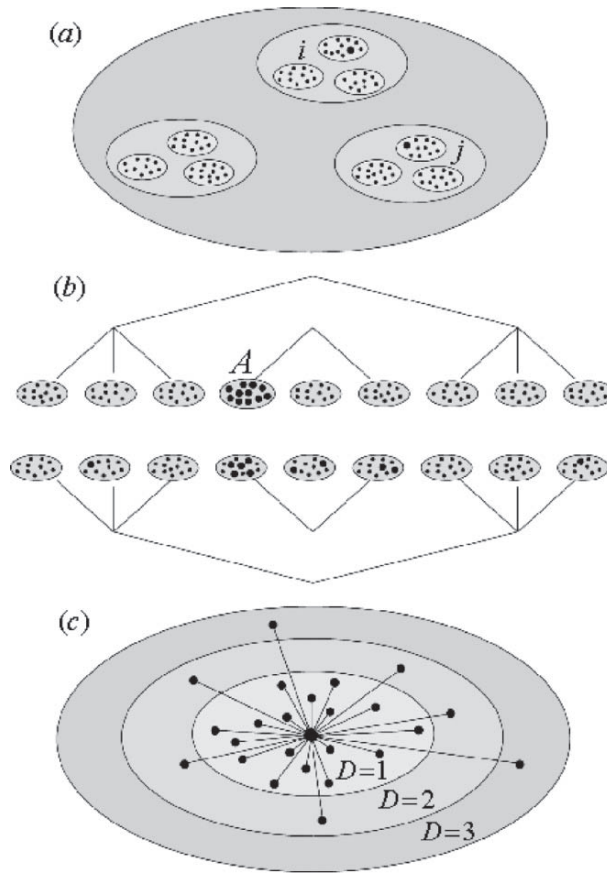


Fig. 2.24. Model of a *social network* (modified and adapted from [MNL03] – see text for explanation).

communication or passage time between nodes, is defined as being the average number of links in the shortest path between a pair of nodes in the network. C represents the degree of local order, and is defined as being the probability that two nodes connected to a common node are also connected to each other.

Many real networks are sparse in the sense that the number of links in the network is much less than $N(N-1)/2$, the number of all possible (bidirectional) links. On one hand, random sparse networks have short average path length (i.e., $L \sim \log N$), but they are poorly clustered (i.e., $C \ll 1$). On the other hand, regular sparse networks are typically highly clustered, but L is comparable to N . (All-to-all networks have $C = 1$ and $L = 1$, so they are most efficient, but most expensive in the sense that they have all $N(N-1)/2$ possible connections and so they are dense rather than sparse.) The small-world network models have advantages of both random and regular sparse

networks: they have small L for fast communication between nodes, and they have large C , ensuring sufficient redundancy for high fault tolerance. Many networks in the real world, such as the world-wide web (WWW) [AJB99], the neural network of *C. elegans* [WS98, Wat99], collaboration networks of actors [WS98, Wat99], and networks of scientific collaboration [New00], have been shown to have this property. The models of small-world networks are constructed from a regular lattice by adding a relatively small number of shortcuts at random, where a link between two nodes u and v is called a *shortcut* if the shortest path length between u and v in the absence of the link is more than two [Wat99]. The regularity of the underlying lattice ensures high clustering, while the shortcuts reduce the size of L .

Most work has focused on average properties of such models over different realizations of *random* shortcut configurations. However, a different point of view is necessary when a network is to be designed to optimize its performance with a restricted number of long-range connections. For example, a transportation network should be designed to have the smallest L possible, so as to maximize the ability of the network to transport people efficiently, while keeping a reasonable cost of building the network. The same can be said about communication networks for efficient exchange of information between nodes.

Most random choices of shortcuts result in a suboptimal configuration, since they do not have any special structures or organizations. On the contrary, many real networks have highly structured configurations of shortcuts. For example, in long-range transportation networks, the airline connections between major cities which can be regarded as shortcuts, are far from being random, but they are organized around hubs. Efficient travel involves ground transportation to a nearest airport, then flights through a hub to an airport closest to the destination, and ground transportation again at the end.

It has been shown in [NML02] that the average path length L of a small-world network with a fixed number of shortcuts attains its minimum value when there exists a ‘center’ node, from which all shortcuts are connected to uniformly distributed nodes in the network (see Figure 2.25(a)). If a small-

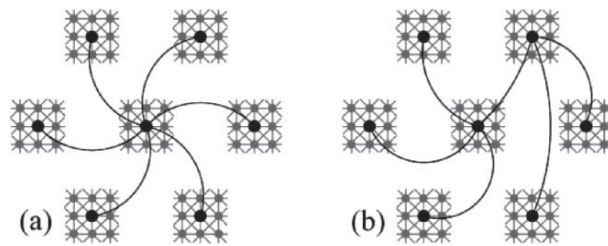


Fig. 2.25. Examples of shortcut configuration with (a) a single center and (b) two centers.

world network has several ‘centers’ and its subnetwork of shortcuts is *connected*, then L is almost as small as the minimum value (see Figure 2.25(b)).

2.3 Intelligent (BDI) Agents Approach

In this section we give a brief review on intelligent (autonomous, BDI) agents approach to brain and cognition. Formally, here we are dealing with an *agent evolution* 2–functor $\mathcal{E}$, given by

$$\begin{array}{ccc}
 \begin{array}{ccc}
 A & \xrightarrow{f} & B \\
 \downarrow h & \text{CURRENT} & \downarrow g \\
 & \text{AGENT} & \\
 \downarrow & \text{STATE} & \\
 C & \xrightarrow{k} & D
 \end{array}
 & \xRightarrow{\mathcal{E}} &
 \begin{array}{ccc}
 \mathcal{E}(A) & \xrightarrow{\mathcal{E}(f)} & \mathcal{E}(B) \\
 \downarrow \mathcal{E}(h) & \text{DESIRED} & \downarrow \mathcal{E}(g) \\
 & \text{AGENT} & \\
 \downarrow & \text{STATE} & \\
 \mathcal{E}(C) & \xrightarrow{\mathcal{E}(k)} & \mathcal{E}(D)
 \end{array}
 \end{array}$$

Here $\mathcal{E}$ represents an association/projection functor from the *source* 2–category of the current agent state, defined as a commutative square of small agent categories $A, B, C, D, \dots$ of *current agent behaviors* and their causal interrelations $f, g, h, k, \dots$, onto the *target* 2–category of the desired agent state, defined as a commutative square of small agent categories $\mathcal{E}(A), \mathcal{E}(B), \mathcal{E}(C), \mathcal{E}(D), \dots$ of *evolved agent behaviors* and their causal interrelations $\mathcal{E}(f), \mathcal{E}(g), \mathcal{E}(h), \mathcal{E}(k)$. As in the previous section, each causal arrow in above diagram, e.g., $f : A \rightarrow B$, stands for a *generic agent dynamorphism*.

Recall that the *agent theory* concerns the definition of the so-called *belief–desire–intention agents* (BDI–agents, for short), as well as multi-agent systems, properties, architectures, communication, cooperation and coordination capabilities (see [RG98]).

A common definition of an agent reads: An agent is a computer system that is situated in some environment, and that is capable of autonomous action in this environment in order to meet its design requirements [Woo00].

Practical side of the agent theory concerns the agent languages and platforms for programming and experimenting with agents. According to [Fer99], a BDI–agent is a physical or virtual entity which:

1. *is capable of limited perceiving its environment* (see Figure 2.26),
2. *has only a partial representation of its environment*,
3. *is capable of acting in an environment*,
4. *can communicate directly with other agents*,
5. *is driven by a set of tendencies*,¹⁸
6. *possesses resources of its own*,

¹⁸ in the form of individual objectives or of a satisfaction/survival function which it tries to optimize

7. *possesses some skills and can offer services,*
8. *may be able to reproduce itself,*
9. *whose behavior tends towards satisfying its objectives,*

– taking into account the resources and skills available to it and depending on its perception, its representation and the communications it receives. Agents' actions affect the environment which, in turn, affects future decisions of agents. The *multi-agent systems* have been successfully applied in numerous fields (see [Fer99] for the review).



Fig. 2.26. A basic agent–environment loop (modified from [Woo00]).

Agents embody a new software development paradigm that attempts to merge some of the theories developed in artificial intelligence research with computer science. The power of agents comes from their intelligence and also their ability to communicate with each other. A simple mapping of agent technology compared to relevant technologies is illustrated in Figure 2.27. Agents can be considered as the successors of *object-oriented programming*

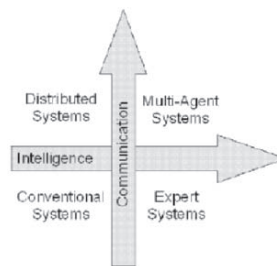


Fig. 2.27. Agent technology compared to relevant technologies.

techniques, applied to certain problem domains. However, the additional layer of implementation in agents provides some key functionalities and deliberately creates a separation between the implementation of an agent from the application being developed. This is done in order to achieve one of the core properties of agents, autonomy. Objects are able to assert a certain amount of control

over themselves via private variables and methods, and other objects via public variables and methods. Consequently, a particular object is able to directly change public variables of other objects and also execute public methods of other objects. Hence, objects have no control over the values of public variables and who and when executes their public methods. Conversely, agents are explicitly separated, and can only request from each other to perform a particular task. Furthermore, it cannot be assumed that after a particular agent makes a request, another agent will do it. This is because performing a particular action may not be in the best interests of the other agent, in which case it would not comply [Woo00].

Types of Intelligent Agents

Here we give a general overview of different types of agents and groups them into several intuitive categories based on the method that they perform their reasoning [Woo00].

Deliberate Agents

Deliberate agents are agents that perform rational reasoning, take actions that are rational after deliberating using their *knowledge base* (KB), carefully considering the possible effects of different actions available to them. There are two subtypes of deliberate agents: *deductive reasoning agents* and *production-rule agents*.

1. Deductive reasoning agents are built using expert systems theory, they operate using an internal symbolic KB of the environment. Desired behavior is achieved by manipulating the environment and updating the KB accordingly. A utility function is implemented that provides an indication on how good a particular state is compared on what the agent should achieve. An example of the idea behind these type of agents is an agent that explores a building. It has the ability to move around and it uses a video camera, the video signal is processed and translated to some symbolic representation. As the agent explores the world it maintains a data structure of what it has explored. The internal structure of deductive reasoning agents is illustrated in Figure 2.28. There are two key problems encountered when trying to build deductive reasoning agents. Firstly, the transduction problem is the problem of translating the real world into an accurate, symbolic description in time for it to be useful. Secondly, the representation or reasoning problem is the problem of representing acquired information symbolically and getting agents to manipulate/reason with it [Woo00].

2. Production systems are also an extension of expert systems. However they place more emphasis how decisions are made based on the state of the KB. The general structure of production system agents is illustrated in Figure 2.29. The KB is called working memory and is aimed to resemble short term memory. They also allow a designer to create a large set of condition-action

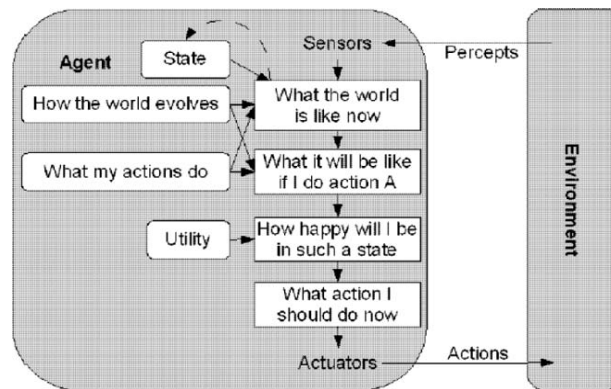


Fig. 2.28. A concept of deductive reasoning agents (modified from [RN03]).

rules called productions that resemble long term memory. When a production is executed it is able cause changes to the environment or directly change the working memory. This in turn possibly activates other productions. Production systems typically contain a small working memory, and a large number of rules that can be executed so fast that production systems are able to operate in real time with thousands of rules [RN03]. An example of a production-rule

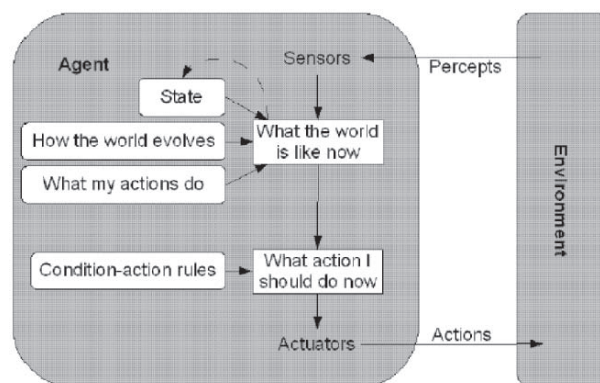


Fig. 2.29. A concept of production-rule agents (modified from [RN03]).

agent development environment is called SOAR (State, Operator And Result). SOAR uses a KB as a problem space and production rules to look for solutions in a problem. IT has a powerful problem solving mechanism whereby every time that it is faced with more than one choice of productions (via a lack of knowledge about what is the best way to proceed) it creates an impasse that results in branching of the paths that it takes through the problem space.

The impasse asserts subgoals that force the creation of sub-states of problem solving behavior with the aim to resolve the super-state impasse [Sio05].

Reactive Agents

Deliberate agents were originally developed using traditional software engineering techniques. Such techniques define pre-conditions required for operation and post-conditions that define the required output after operation. Some agents however, cannot be easily developed using this method because they maintain a constant interaction with a dynamic environment, hence they are called reactive agents. Reactive agents are especially suited for real-time applications where there are strict time constraints (i.e., milliseconds) on choosing actions.

Reactive systems are studied by behavioral means where researchers have tried to use entirely new approaches that reject any symbolic representation and decision making. Instead, they argue that intelligent and rational behavior emerges from the interaction of various simpler behaviors and is directly linked to the environment that the agent occupies [Woo00]. The general structure of reactive agents is illustrated in Figure 2.30. The main contributor of reactive

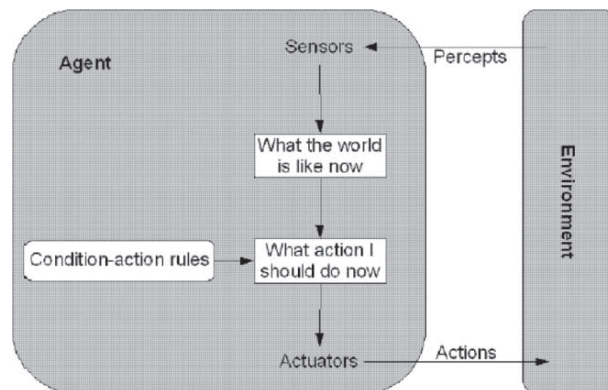


Fig. 2.30. A concept of reactive agents (modified from [RN03]).

agent research is Rod Brooks from MIT, with his *subsumption architecture*, where decision making is realized through a set of task-accomplishing behaviors. Behaviors are arranged into layers where lower layers have a higher priority and are able to inhibit higher layers that represent more abstract behaviors [Bro86]. A simple example of the subsumption architecture is a multi-agent system used to collect a specific type of rock scattered in a particular area on a distant planet. Agents are able to move around, collect rocks and return to the mother-ship. Due to obstacles on the surface of the planet, agents are not able to communicate directly, however they can carry special

radioactive crumb that they drop on the ground for other agents to detect. The crumbs are used to leave a trail for other agents to follow. Additionally, a powerful locator signal is transmitted from the mother-ship, agents can find the ship by moving towards a stronger signal. A possible behavior architecture for this scenario are the following set of *heuristic IF-THEN rules*:

1. IF detect an obstacle THEN change direction (this rule ensures that the agent avoids obstacles when moving);
2. IF carrying samples and at the base THEN drop samples (this rule allows agent to drop samples in the mother-ship);
3. IF carrying samples and not at the base THEN drop 2 crumbs and travel up signal strength (this rule either reinforces a previous trail or creates a new one);
4. IF detect a sample THEN pick sample up (this rule collects samples);
5. IF sense crumbs THEN pick up 1 crumb and travel away from signal strength (this rule follows a crumb trail that should end at a mineral deposit; crumbs are picked up to weaken the trail such that it disappears when the mineral deposit has depleted);
6. IF true THEN move randomly (this rule explores the area until it stumbles upon a mineral deposit or a crumb trail).

Hybrid Agents

Hybrid agents are capable of expressing both reactive and pro-active behavior. They do this by breaking reactive and proactive behavior into different subsystems called layers. The lowest layer is the reactive layer and it provides immediate responses to changes for the environment, similarly to the subsumption architecture. The middle layer is the planning layer that is responsible for telling the agent what to do by reviewing internal plans, and selecting a particular plan that would be suitable for achieving a goal. The highest layer is the modeling layer that manages goals. A major issue encountered when developing solutions with hybrid reasoning agents is that agents must be able to balance the time spent between thinking and acting. This includes being able to stop planning at some point and commit to goal, even if that goal is not optimal [Woo00]. The general structure of hybrid agents is illustrated in Figure 2.31.

Agent-Oriented Software Development

Agent-oriented development is concerned with the techniques of software development that are specifically suited for developing agent systems. This is an important issue because existing software development techniques are unsuitable for agents as there exists a fundamental mismatch between traditional software engineering concepts and agents. Consequently, traditional techniques fail to adequately capture an agent's autonomous problem-solving

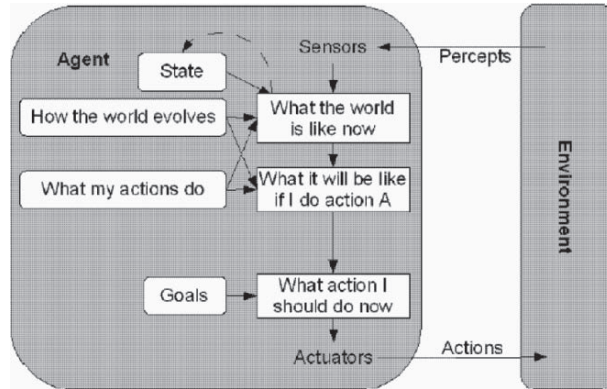


Fig. 2.31. A concept of hybrid, goal-directed agents (modified from [RN03]).

behavior as well as the complex issues involved in multi-agent interaction [Sio05].

The first agent-oriented methodology was proposed by Wooldridge and is called Gaia. Gaia is deemed appropriate for agent systems with the following characteristics: (i) Agents are smart enough to require significant computational resources. (ii) Agents may be implemented using different programming languages, architectures or techniques. (iii) The system has a static organization structure such that inter-agent relationships do not change during operation. (iv) The abilities of agents and the services they provide do not change during operation. (v) The system requires only small amount of agents. Gaia splits the development process into three phases: Requirements, Analysis and Design. The requirements phase is treated in the same way as traditional systems. The analysis phase is concerned with the roles that agents play in the system as well as the interactions required between agents. The design phase is concerned with the agent types that will make up the system. The agent main services that are required to realize the agent's roles, and finally, the lines of communication between the different agents. The Gaia methodology was the inspiration for the more detailed methodology described in the next section (see [Woo00]).

2.3.1 Agents Environments

Agent technology has been applied to many different application areas, each focusing on a specific aspect of agents that is applicable to the domain at hand. The role that BDI-agents play in their environment distinctly depends on the application domain. The agent research community is very active and environments are mostly viewed as test-beds for developing new features in agents and showing how they are successfully used to solve a particular problem. Fortunately, in most cases this is a two-sided process, by understanding, developing and improving new agent technologies it becomes possible to solve

similar real life problems. Consequently, as the underlying foundation of agent software matures, new publications describe how agents are being applied successfully in increasingly complex application domains [Sio05].

The BDI-agent is usually understood to be a *decision-maker* and anything that it interacts with, comprising everything outside the agent itself, is referred to as the *environment*. The environment has a number of features and generates *sensations* that contain some information about the features. A *situation* is commonly understood as a complete snapshot of the environment for a particular instance in time.¹⁹ Hence, if an agent is able to get or deduce the situation of its environment it would know everything about the environment at that time. A *state* is here defined as a snapshot of the agent's beliefs corresponding to its limited understanding of the environment. This means that the state may or may not be a complete or accurate representation of the situation. This distinction supports research being conducted on improving the agent's *situation awareness* (SA), whereby SA measures how similar the state is as opposed to the situation.

The agent and the environment interact continually, the agent selects actions and the environment responds to the actions by presenting new sensations to the agent [SB98]. The interaction is normally segmented in a sequence of discrete time steps, whereby, at a particular time step the agent receives data from the environment and on that basis selects an action. In the next time step, the agent finds itself in a new state (see Figure 2.26).

Various properties of environments have been classified into six categories [RN03]:

1. *Fully observable or partially observable.* A fully observable environment provides the agent with complete, accurate and up-to-date information of the entire situation. However, as the complexity of environments increases, they become less and less observable. The physical world is considered a partially observable environment because it is not possible to know everything that happens in it [Woo00]. On the other hand, depending on the application, the environment should not be expected to be completely observable (e.g., if an agent is playing a card game it should not be expected to know the cards of every other player). Hence, in this case, even though there is hidden information in the environment and this information would be useful if the agent knew it, is not necessary for making rational decisions [SB98]. An extension of this property is when sensations received from the environment are able to summarize past sensations in a compact way such that all relevant information from the situation can be deduced. This requires that the agent maintains a history of all past sensations. When sensations succeeds in retaining all relevant information, they are said to have the Markov property. An example of a Markov sensation for

¹⁹ In a number of references, the term state is used with the same meaning. In this section a clear distinction is made between the two terms, a situation is defined as a complete snapshot of the real environment.

a game of checkers is the current configuration of the pieces on the board, this is because it summarizes the complete sequence of sensations that led to it. Even though much of the information about the sequence is lost, all important information about the future of the game is retained. A difficulty encountered when dealing with partially observable environments is when the agent is fooled to perceiving two or more different situations as the same state, this problem is known as perceptual aliasing. If the same action is required for the different situations then aliasing is a desirable effect, and can be considered a core part of the agent's design, this technique is commonly called state generalization [SB98].

2. *Deterministic or stochastic.* Deterministic is the property when actions in the environment have a single guaranteed effect. In other words, if the same action is performed from the same situation, the result is always the same. A useful consequence of a deterministic environment is the ability to predict what will happen before an action is taken, giving rise to the possibility of evaluating multiple actions depending on their predicted effects. The physical world is classified as a stochastic environment as stated by [Woo00]. However, if an environment is partially observable it may appear to be stochastic because not all changes are observed and understood [RN03], if more detailed observations are made, including additional information, the environment becomes increasingly deterministic.
3. *Episodic or sequential.* Within an episodic environment, the situations generated are dependent on a number of distinct episodes, and there is no direct association between situations of different episodes. Episodic environments are simpler for agent development because the reasoning of the agent is based only on the current episode, there is no reason to consider future episodes [Woo00]. An important assumption made when designing agents for episodic environments, is that all episodes eventually terminate no matter what actions are selected [SB98]. This is particularly true when using learning techniques that only operate on the completion of an episode through using a captured history of situations that occurred within the episode. Actions made in sequential environments, on the other hand, affect all future decisions. Chess is an example of a sequential environment because short-term actions have long-term consequences.
4. *Static or dynamic.* A static environment is one that remains unchanged unless the agent explicitly causes changes through actions taken. A dynamic environment is one that contains other entities that cause changes in ways beyond the agents control. The physical world continuously changes with external means and is therefore considered a highly dynamic environment [Woo00]. An example of a static environment, is an agent finding its way through a 2D maze. In this case all changes are caused by the same agent. An advantage of static environments is that the agent does not need to continuously observe the environment while its deciding the next action. It can take as much time as it needs to make a decision and the environment will be the same as when previously observed [RN03].

5. *Discrete or continuous.* An environment is discrete if there is a fixed, finite number of actions and situations in it [Woo00]. Simulations and computer games are examples of discrete environments because they involve capturing actions performed by entities, processing the changes caused by the actions and providing an updated situation. Sometimes however, this process is so quick that the simulation appears to be running continuously. An example of a continuous environment is taxi driving, because the speed and location of the taxi and other cars changes smoothly over time [RN03].
6. *Single-agent or multi-agent.* Although the distinction between single and multi-agent environments may seem trivial, recent research has surfaced some interesting issues. These arise from the question of what in the environment may be viewed as another agent [RN03]. For example, does a taxi driver agent need to treat another car as an agent? What about a traffic light or a road sign? An extension to this question is when humans are included as part of the design of the system, giving rise to the new research area called human-agent teaming [Sio05].

2.3.2 Agents' Reasoning and Learning

The environments described above illustrate the need for *adaptation* when agent systems are required to interact with complex environments. Here we will review how agents and humans are understood to perform reasoning and learning when they are faced with a particular environment.

Reasoning is understood as the thinking process that occurs within an agent that needs to make a particular decision. This topic has been tackled via two parallel directions with two different schools of thought. The first school of thought focuses on how agents can perform rational reasoning where the decisions made are a direct reflection of knowledge. The advantage of this approach is that decisions made by an agent can be understood simply by looking within its internal data structures, as the agent only makes decisions based on what it knows. This process includes maintaining the agent's knowledge base such that it contains accurate information about its environment, by performing operations in order to keep all knowledge consistent. Decisions are made through a collection of rules applied on the knowledge base that define what should occur as knowledge changes [Sio05].

Another school of thought is concerned with the way that humans perform reasoning and apply any concepts developed to agent technology. Humans are known to perform practical reasoning every day, their decisions are based on their desires and their understanding in regards to how to go about achieving them. The process that takes place between observing the world, considering desires and taking actions can be broken up into four main stages, each of which consists of a number of smaller components. Through learning, it also becomes possible to create agents that are able to change the way that they were originally programmed to behave. This can be advantageous when an

agent is faced with a situation that it does not know how to proceed. Furthermore, it is useful when an agent is required to improve its performance with experience.

Reasoning and Behavior

Research on artificial reasoning and behavior has been tackled from different angles that can be categorized along two main dimensions (see Figure 2.32). The vertical dimension illustrates the opposing nature of reasoning and behavior that correspond to thinking versus acting respectively. This is an important feature concept in every application using AI techniques. Great emphasis is given to the balance between processing time for making better decisions, and the required speed of operation. Approaches falling to the left side are based on how humans reason and behave while approaches falling on the right side are concerned with building systems that are rational, meaning that they are required to think and act as best they can, given their limited knowledge [RN03].

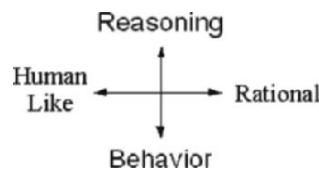


Fig. 2.32. Reasoning dimensions (modified from [RN03]).

Rational Reasoning

Representation and search. Recall that the way that information is represented and used for intelligent problem solving forms a number of important but difficult challenges that lie within the core of AI research. Knowledge representation is concerned with the principles of correct reasoning. This involves two parallel topics of research. One side is concerned with the development of formal representation languages with the ability to maintain consistent knowledge about the world, the other side is concerned with the development of reasoning processes that bring the knowledge to life. The output of both of these areas results in a Knowledge Base (KB) system. KBs try to create a model of the real world via the collection of a number of sentences. An agent is normally able to add new sentences to the knowledge base as well as query the KB for information. Both of these tasks may require the KB to perform inference on its knowledge, where an inference is defined as the process of deriving new sentences from known information. An additional requirement of KBs is that when an agent queries the KB, the answer should be inferred

from information previously added to the KB and not from unknown facts. The most important part of a KB is the logic in which the its sentences are represented. This is because all sentences in a KB are in fact expressed according to the syntax and semantics of the logic's representation language. The syntax of the logic is required for implementing well formed sentences while the semantics define the truth of each sentence with respect to a model of the environment being represented [RN03].

Problem solving using KBs involves the use of search algorithms that are able to search for solutions between different states of information within the KB. Searching involves starting from an initial state and expanding across different successor state possibilities until a solution is found. When a search algorithm is faced with a choice of possibilities to consider, each possibility is thoroughly searched before moving to the next possibility. Search however has a number of issues, including [Lug02]:

(i) Guarantee of a solution being available; (ii) Termination of the search algorithm; (iii) The optimality of a particular solution found; and (iv) The complexity of the search algorithm with respect to the time and memory usage.

State space analysis is done with the use of graphs. A graph is a set of nodes with arcs that connect them, each node can have a label to distinguish it from another node and arcs can have directions to indicate the direction of movement between the nodes. A path in the graph connects a sequence of nodes with arcs and the root is a node that has a path to all other nodes in the graph.

There are two ways to search a state space, the first way is to use data-driven search by which the search starts by a given set of facts and rules for changing states. The search proceeds until it generates a path that leads to the goal condition. Data driven search is more appropriate for problems in which the initial problem state is well defined, or there are a large number of potential goals and only a few facts to start with, or the goal state is unclear [Lug02].

The second way is to use goal-driven search by which the search starts by taking the goal state and determining what conditions must be true to move into the goal state. These conditions are then treated as subgoals to be searched. The search then continues backwards through the subgoals until it reaches the initial facts of the problem. Goal driven search is more appropriate for problems in which the goal state is well defined, or there are a large number of initial facts making it impractical to prefer data driven search, or the initial data is not given and must be acquired by the system [Lug02].

The choice of which of the options to expand first is defined by the algorithm's search strategy. Two well known search strategies are: Breadth-first, where all successors of a given depth are expanded first before any nodes at the next level. Depth-first search involves expanding the deepest node for a particular option before moving to the next option. There are also strategies that include both elements, for example defining a depth limit for searching in

a tree. It is also possible to use heuristics to help with choosing branches that are more likely to lead to an acceptable solution. Heuristics are usually applied when a problem does not have an exact solution or the computational cost to find an exact solution is too big. They reduce the state space by following the more promising paths through the state space [RN03].

An additional layer of complexity in knowledge representation and search is due to the fact that agents almost never have access a truly observable environment. Which means that agents are required to act under *uncertainty*. There are two techniques that have been used for reasoning in uncertain situations. The first involves the use of probability theory in assigning a value that represents a degree of belief in facts in the KB. The second method involves the use of *fuzzy sets* (see below) for representing how well a particular object satisfies a vague description [RN03].

Expert systems. Recall that knowledge-based reasoning systems are commonly called *expert systems* because they work by accumulating knowledge extracted from different sources, and use different strategies on the knowledge in order to solve problems. Simply put, expert systems try to replicate what a human expert would do if faced with the same problem. They can be classified into different categories depending on the type of problem they are used to solve [Lug02]:

- *interpretation*: making conclusions or descriptions from collections of raw data;
- *prediction/forecasting*: predicting the consequences of given situations;
- *diagnosis*: finding the cause of malfunctions based on the symptoms observed;
- *design*: finding a configuration of components that best meets performance goals when considering several design constraints;
- *planning*: finding a sequence of actions to achieve some given goals using specific starting conditions and run-time constraints;
- *monitoring*: observing a system’s behavior and comparing it to its expected behavior at run-time;
- *debugging*: finding problems and repairing caused malfunctions; and
- *control*: controlling how a complex system behaves.

A common way to represent data in a expert systems is using first-order predicate calculus formulae. For example, the sentence ‘If a bird is a crow then it is black’ is represented as:

$$\forall X(crow(X) \implies black(X)).$$

Production systems are based on a model of computation that uses search algorithms and models human problem solving. Production systems consist of production rules and a working memory. Production rules are pre-defined rules that describe a single segment of problem-solving knowledge. They are represented by a condition that determines when the production

is applicable to be executed, and an action which defines what to do when executed. The working memory is an integrated KB that contains an ever-changing state of the world.

The operation of production systems generally follows a *recognize-act cycle* (see Figure 2.33). Working memory is initialized with data from the initial problem description and is subsequently updated with new information. At every step of operation, the state presented by the working memory is continuously captured as patterns and applied to conditions of productions. If a pattern is recognized against a condition, the associated production is added to a conflict set. A conflict resolution operation chooses between all enabled productions and the chosen production is fired by executing its associated action. The actions executed can have two effects. Firstly, they can cause changes to the agent's environment which indirectly changes the working memory. Secondly, they can explicitly cause changes in the working memory. The cycle then restarts using the modified working memory until a situation when no subsequent productions are enabled. Some production systems also contain the means to do backtracking when there are no further enabled productions but the goal of the system has still not been reached. Backtracking allows the system to work backwards and try some different options in order to achieve its goal [Lug02].

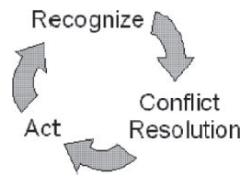


Fig. 2.33. A recognize-act operation cycle of production systems (modified from [Lug02]).

Human Reasoning

Practical reasoning is concerned with studying the way that humans reason about what to do in everyday activities and applying this to the design of intelligent agents. Practical reasoning is specifically geared to reasoning towards actions, it involves weighing conflicting considerations of different options that are available depending on what a person desires to do. Practical reasoning can be divided into two distinct activities (see Figure 2.34). The first activity is called *deliberation reasoning*, it involves deciding on what state to achieve. The second activity is called *means-ends reasoning* and it involves deciding on how to achieve this state of affairs [Woo00]. Recall that the central component of practical reasoning is the concept of *intention* because it is used to characterize both the action and thinking process of a person. For example ‘in-

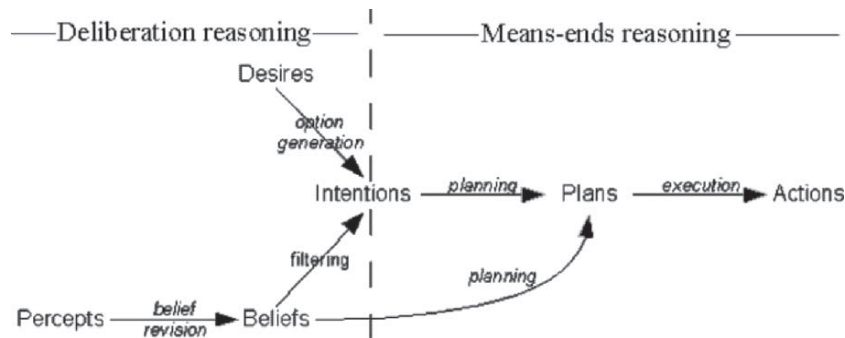


Fig. 2.34. BDI-reasoning process (modified from [Woo00]).

tending to do something’ characterizes a persons thinking while ‘intentionally doing something’ characterizes the action being taken.

The precursors of an intention are a persons’s desires and beliefs and hence all of the beliefs, desires and intentions must be consistent. In other words, intending to do something must be associated with a relevant desire, as well as the belief that the intended action will help to achieve the desire. Maintaining this consistency is challenging due to the dynamic nature of desires and beliefs. Desires are always changing according to internal self-needs while beliefs are constantly updated using information obtained from senses through a process called belief revision, from the external environment.

Forming an intention involves performing two concurrent operations. Firstly, option generation uses the current desires to generate a set of possible alternatives. Secondly, filtering chooses between these alternatives based on the current intentions and beliefs. An intention also requires assigning a degree of commitment toward performing a particular action or set of actions in the future. There are four important characteristics emerging by this commitment are [Woo00]:

1. Intentions drive means-ends reasoning by forcing the agent to decide on how to achieve them.
2. Intentions persist by forcing a continuous strive to achieve them. Hence, after a particular action has failed, other alternative actions are attempted until it comes to be believed that it is not possible to achieve the intention, or the relevant desire is not longer present.
3. Intentions constrain future deliberation because it is not necessary to consider desires that are inconsistent with the current intentions.
4. Intentions influence beliefs by introducing future expectations. This is due the requirement of believing that a desired state is possible before and during execution the intention to satisfy it.

The process that occurs after forming an intention in order to take action is identified as planning, it involves selecting and advancing through a sequence

of plans that dictate what actions to take. Plans are understood to consist of pre-condition that characterizes the state in which a plan is applicable for execution and a post-condition characterizes the resulting state after executing the plan. Finally, a body containing the recipe defining the actions to take [Woo00]. From the theory of practical reasoning, researchers have been able to develop intuitive agent development architectures. The transition between the theory and implementation has required the identification of equivalent software constructs for each of the BDI-components [Sio05].

Cognitive systems engineering takes into account, during the design and implementation of systems, that systems will be used by humans. It acknowledges that humans are dynamic entities that are part of the system itself but cannot be modeled as static components of a system. When humans use a system they adapt to the functional characteristics of the system. In addition, sometimes they can modify the system's functional characteristics in order to suit their own needs and preferences. This means that in order to understand the behavior of the system once the adaptation has happened is to abstract the structural elements into a purely functional level and identify and separate the functional relationships. This concept can best be understood using a simple example from [RPG94]:

“When a novice is driving a car, it is based on an instruction manual identifying the controls of the car and explaining the use of instrument readings, that is, when to shift gears, what distance to maintain to the car ahead (depending on the speed), and how to use the steering wheel. In this way, the function of the car is controlled by discrete rules related to separate observations, and navigation depends on continuous observation of the heading error and correction by steering wheel movements. This aggregation of car characteristics and instructed input-output behavior makes it possible to drive; it initiates the novice by synchronizing them to the car functions. However, when driving skill evolves, the picture changes radically. Behavior changes from a sequence of separate acts to a complex, continuous behavioral pattern. Variables are no longer observed individually. Complex patterns of movements are synchronized with situational patterns and navigation depends on the perception of a field of safe driving. The drivers are perceiving the environment in terms of their driving goals. At this stage, the behavior of the system cannot be decomposed into structural elements. A description must be based on abstraction into functional relationships.”

A new design approach is introduced that shifts away from the traditional software engineering perspective to a functional perspective. There are two different ways to define functional characteristics. Firstly, relational representations are based on mathematical equations that relate physical, measurable environments. Secondly, casual representations are connections between different events. [RPG94] presented a framework that made it possible to relate

conceptual characteristics. The framework takes into account that in order to bridge system behaviors into human profiles and preferences, several different perspectives of analysis and languages of representation are needed (see Figure 2.35).

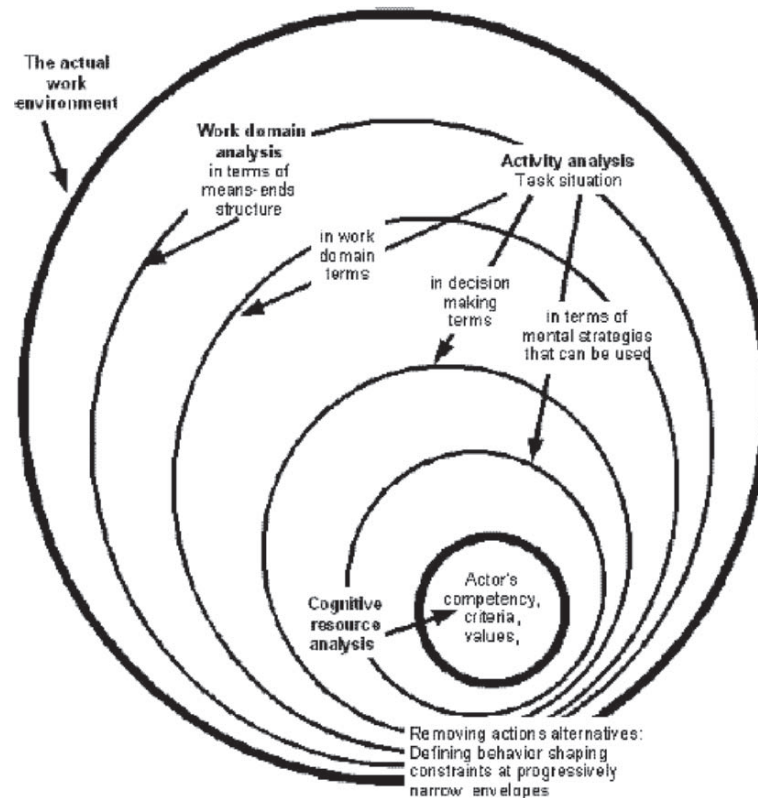


Fig. 2.35. Relating Work Environment to Cognitive Resource Profiles of Actors (adapted from [RPG94]).

In this framework, the *work domain analysis* is used to make explicit the goals, constraints and resources found in a work system. They are represented by a general inventory of system elements that are categorized by functional elements and their means-ends relations. The analysis identifies the structure and general content of the global knowledge of the work system. Activity analysis is divided into three different dimensions. Firstly, activity analysis in domain terms focuses on the freedom left for activities after the constraints posed by time and the functional space of the task. Generalizations are made in terms of objectives, functions and resources. Secondly, activity analysis in decision terms use functional languages to identify decision making functions

within relevant tasks. This results of this analysis are used to identify prototype knowledge states that connect different decision functions together. Thirdly, mental strategies are used to compare task requirements with cognitive resource profiles of the individual actors and how they perform their work, thus supplies the designer with mental models, data formats and rule sets that can be incorporated into the interface of the system and used by actors of varying expertise and competence.

The *work organization analysis* is used to identify the actors involved in the decisions of different situations. This is done by finding the principles and criteria that govern the allocation of roles among the groups and group members. This allocation is dynamically dependent on circumstances and is governed by different criteria such as actor competency, access to information, minimizing communication load and sharing workload.

The *social organization analysis* focuses on the social aspect of groups working together. This is useful for understanding communication between team members, such communication may include complex information like intentions used for coordinating activities and resolving ambiguities or misinterpretations. Finally, User Analysis is used to help judge which strategy is likely to be chosen by an actor in a given situation focusing on the expertise and the performance criteria of each actor.

Rasmussen further proposes a framework for representing the various states of knowledge and information processes of human reasoning, it is called the *decision ladder* (see Figure 2.36). The ladder models the human decision making process through a set of generic operations and standardized key nodes or states of knowledge about the environment. The circles illustrated are states of knowledge and the squares are operations. The decision ladder was developed as a model for performing work domain analysis, however, the structure of the ladder is generic enough to be used as a guide in the context of describing agent reasoning. The decision ladder can be further segmented into three levels of expertise [RPG94]. The skill (lowest) level represents very fast, automated sensory-motor performance and it is illustrated in the ladder via the heuristic shortcut links in the middle. The rule (medium) level represents the use of rules and/or procedures that have been pre-defined, or derived empirically using experience, or communicated by others, it traverses the bottom half of the ladder. Finally, the knowledge (highest) level represents behaviors during less-familiar situations when someone is faced with an environment where there are no rules or skills available, in such cases a more detailed analysis of the environment is required with respect to the goals the agent is trying to achieve, the entire ladder is used for this case.

Learning and Adaptation

The concept of learning is vague and has many definitions, this is because learning is used for problem solving in many different areas. Arguably learning should have the following features [Sio05]:

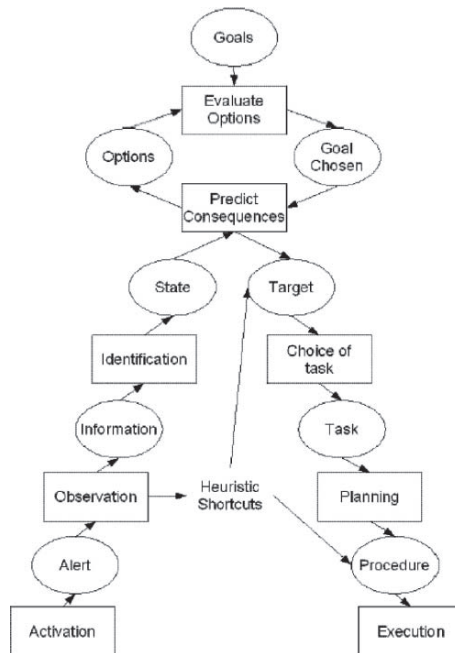


Fig. 2.36. Rassmusen's *decision ladder* (adapted from [RPG94]).

1. Should be a process of obtaining skill where 'skill' means a pre-existing program that produces useful behavior.
2. Produces alteration of an individual behavior where 'alteration' implies that the program that produces the behavior should be changed.
3. Produces a change in a behavioral potentiality where 'potentiality' refers to a storage of the learning program.
4. Develops an inner program that is better adapted to its task where 'better' implies a measurement should be built into the learning mechanism.
5. It is enabled such that it can perform a task more efficiently where 'efficiency' implies that the performance should be measured.
6. It changes the quality of the output behavior where 'quality' implies that the output behavior should be measured.
7. Making useful changes in mind where 'useful' refers to measurement and 'mind' refers to the ability to store and modify the learning program.
8. Constructing and modifying representations where 'constructing' implies the ability to construct new learning programs.
9. Learning is essentially discovery (creating new programs).
10. Formation of new classes and categories via generalization.
11. Changing the algorithm where 'algorithm' means the learning program.
12. Forcing the system to have a particular response to a specific signal by repeating the input signals.

13. Evolving the model of the world.

According to [Lug02], learning is a change in a system that allows it to perform better on the second time when a task of the system is repeated, or on a similar task. The main problem here is selecting the changes to a system that will change the performance of the system. An extension to this problem is changing system parameters in a way that will improve and not hinder the system performance. The above is a broad description of what learning is and as it can be seen that the concept of learning is not trivial. So the next question is: What technologies are already available that attempt to introduce some learning into systems?

Machine Learning

Symbol-based learning relies on learning algorithms that can be characterized into the following five dimensions [Lug02]:

- *data and goals*: here the learning problem is described according to the goals of the learner and the data it is initially given;
- *knowledge representation*: using representation languages with programs to store the knowledge learned by the system in a logical way;
- *learning operations*: an agent is given a set of training instances and it is tasked to construct a generalization, heuristic rule or a plan that satisfies its goals;
- *concept space*: the representation language along with the learning operations define a space of possible concept definitions, the learner needs to search this space to find the desired concept. The complexity of this concept space is used to measure how difficult the problem is; and
- *heuristic search*: heuristics are used to commit to a particular direction when searching the concept space.

Learning algorithms can be grouped as listed below. Each dimension focuses on a different sub-area of symbol-based learning [Lug02].

Connectionist learning is performed using artificial neural networks (see section below), which are systems comprised of a large number of interconnected artificial neurons. They have been widely used for (see, e.g., [Hay94, Kos92, Lug02]):

- *classification*: deciding the category or grouping where an input value belongs;
- *pattern recognition*: identifying a structure in sometimes noisy data;
- *memory recall*: addressing the content in memory;
- *prediction/forecasting*: identifying an effect from different causes;
- *optimization*: finding the best organization within different constraints; and
- *noise filtering*: separating a signal from the background noise or removing irrelevant components to a signal.

The knowledge of the network is encapsulated within the organization and interaction of the neurons. Specifically, the global properties of neurons are characterized as:

- *network topology*: the topology of the network is the pattern of connections between neurons;
- *learning algorithm*: the algorithm used to change the weight between different connections; and
- *encoding scheme*: the interpretation of input data presented to the network and output data obtained from the network.

Learning is achieved by modifying the structure of the neural network, via adjusting weights, in order to map input combinations to required outputs. There are two general classes of learning algorithms for training neural networks, they are supervised and unsupervised learning. Supervised learning requires the neural network to have a set of training data, consisting of the set of data to be learned as well as the corresponding answer. The data set is repeatedly presented to the neural network, in turn, the network adapts by changing the weights of connections between the neurons until the network output corresponds closely to the required answers. The goal of supervised learning is to find a model or mapping that will correctly associate its inputs with its targets. Supervised learning is suited to applications when the outputs expected from the network are well known. This allows the designer (or another fully trained network) to provide feedback.

In the case of unsupervised learning the target value is not provided and the information in the training data set is continuously presented until some convergence criteria is satisfied. This involves monitoring the output of the network and stopping its training when some desired output is observed. The main difference to supervised learning is that the desired output is not known when the training starts. During training, the network has to continuously adapt and change its output until it demonstrates a useful output behavior at which time it receives a single feedback to stop. The input data provided to the network will need to include sufficient information so that the problem is unambiguous. Unsupervised learning is suitable in situations where there is no clear-cut answer to a given problem.

The biggest problem of using neural networks with agents with that the concepts cannot intuitively fit within the agent oriented paradigm. However, neural networks have been used to implement part of a system such as pattern recognition and classification. It is also believed that neural learning concepts and techniques will play an important role in future research [Lug02].

Social and emergent learning focuses on learning algorithms using the underlying concept of evolution, in other words, shaping a population $P(t)$ of candidate solutions x_i^t through the survival of the fittest members at time t . $P(t)$ is defined as:

$$P(t) = \{x_1^t, x_2^t, \dots, x_n^t\}.$$

The attributes of a solution are represented with a particular pattern that is initialized by a *genetic algorithm*. As time passes, solution candidates are evaluated according to a specific fitness function that returns a measure of the candidate's fitness at that time. After evaluating all candidates the algorithm selects pairs for recombination. Genetic operators from each individual are used to produce new solutions that combine components of their parents. The fitness of a candidate determines the extent to which it reproduces. The general form of the genetic algorithm reads [Lug02]:

1. $t \leftarrow 0$;
2. Initialize population $P(t)$;
3. **while** termination condition not met **do**;
4. **for** each member x_i^t within $P(t)$ **do**;
5. $fitness(member) \leftarrow FitnessFunction(member)$;
6. **end for**;
7. select members from $P(t)$ based on $fitness(member)$;
8. produce offspring of selected members using generic operators;
9. replace members of $P(t)$ with offspring based on fitness;
10. $t \leftarrow t + 1$;
11. **end while**.

Reinforcement Learning

Recall that *reinforcement learning* (RL) is designed to allow computers to *learn by trial and error*. It is an approach to machine intelligence that combines two disciplines to solve a problem that each discipline cannot solve on its own. The first discipline, *dynamic programming* is a field in mathematics used to solve problems of optimization and control. The second discipline, supervised learning is discussed in section on neural networks below. In most real-life problems the correct answers required with supervised learning are not available, using RL the agent is simply provided with a *reward-signal* that implicitly trains the agent as required, Figure 2.37 illustrates the agent-environment interaction used with RL. The agent and the environment interact in a discrete sequence of time steps $t = 0, 1, 2, 3, \dots$, for each time step the agent is presented with the current instance of the state $s_t \in S$ where S is the set of all possible states. The agent then uses the state to select and execute an action $a_t \in A(st)$ where $A(st)$ is the set of all possible actions available in state st . In the next time step the agent receives a reward $r_{t+1} \in R$, and is presented with a new state s_{t+1} . The system learns by mapping an action to each state for a particular environment. A specific mapping of actions and states is known as a *policy* π where $\pi_t(s, a)$ is the probability that $a_t = a$ if $s_t = s$. Actions available to agents can be separated into three different categories [SB98]:

- Low-level actions (e.g., supplying voltage to a motor);
- High-level actions (e.g., making a decision);

- Mental actions (e.g., shifting attention focus);



Fig. 2.37. The agent–environment interface in reinforcement learning (adapted from [SB98]).

An important point to note is that according to Figure 2.37, the reward is calculated by the environment which is external to the agent. This is a confusing concept because at first it seems that the designer of an RL system is required to somehow implement something in the environment in order to provide an agent with appropriate rewards. The RL literature overcome this problem by explaining that the boundary between the agent and the environment need not be distinctively physical. The boundary of the agent is shortened to include only the reasoning process, everything outside the reasoning process which includes all other components of the agent, are treated as part of the environment. In the context of human reasoning, this is analogous to treating the human brain as the agent and the entire human body as part of the environment [Sio05].

Markov property of RL is concerned with the way that the state signal received from the environment is represented. This is an important issue when developing an RL system because all actions are directly dependent on the state of the environment. In a causal system the response of the environment for an action taken at time t will depend on all actions previously taken, formally written as

$$PR\{s_{t+1} = s', \quad r_{t+1} = r | s_t, a_t, r_t, s_{t-1}, a_{t-1}, r_{t-1}, \dots, s_0, a_0\}.$$

However, the state signal should not be expected to represent everything about the environment because certain information might be inaccessible or intentionally made unavailable.

When the response of the environment depends only on the state and action representations at time t , is it said to have the Markov property and can be defined as

$$PR\{s_{t+1} = s', \quad r_{t+1} = r | s_t, a_t\}.$$

This means that the state signal is able to summarize all past sensations compactly such that all relevant information is retained for making decisions.

When a reinforcement learning problem satisfies the Markov property it is called a *Markov decision process* (MDP), additionally if the states and actions sets are finite then it is called a finite MDP. In some cases even when

a particular problem is non-Markov it may be possible to consider it as an approximation of an MDP for the basis for learning, in such cases the learning performance will depend on how good the approximation is.

Reward function $R_{ss'}^a$ provides rewards depending on the actions of the agent. The sequence of rewards received after time step t is $r_{t+1}, r_{t+2}, r_{t+3}, \dots$, the agent learns by trying to maximize the sum of rewards received when starting from an initial state and proceeding to a terminal state. An additional concept is the one when an agent tries to maximize the expected discounted return as

$$R_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \dots = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1},$$

where $0 \leq \gamma \leq 1$. This involves the agent discounting future rewards by a factor of γ .

There are two important classes of reward functions [HH97]. In the *pure delayed reward functions*, rewards are all zero except at a terminal state where the sign of the reward indicates whether it is a goal or penalty state. A classic example of pure delayed rewards is the cart-pole problem, where the cart is supporting a hinged inverted pendulum and the goal of the RL agent is to learn to balance the pendulum in an upright position. The agent has two actions in every state, move left and move right. The reinforcement function is zero everywhere except when the pole falls or the cart hits the end of the track, when the agent receives a -1 reward. Through such a set-up an agent will eventually learn to balance the pole and avoid the negative reinforcement. On the other hand, using the *minimum-time reward functions* it becomes possible to find the shortest path to a goal state. The reward function returns a reward of -1 for all actions except for the one leading to a terminal state for which the value is again dependent on whether it is a goal or penalty state. Due to the fact that the agent wants to maximize its rewards, it tries to achieve its goal at the minimum number of actions and therefore learns the optimal policy. An example used to illustrate this problem is driving a car up the hill problem, which is caused by the car not having enough thrust to drive up the hill on its own and therefore the RL agent needs to learn to use the momentum of the car climb the hill.

Value function. The issue of how an agent knows what is a good action is tackled using the *value function* $V^\pi(s)$ which provides a value of ‘goodness’ to states with respect to a specific policy. For MDPs, the information in a value function can be formally defined by

$$V^\pi(s) = E_\pi\{R_t | s_t = s\} = E_\pi\left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t = s\right\}$$

where $E_\pi\{\}$ denotes the expected value if the agent follows policy π , this is called the *state value function*. Similarly, the *action value function* starting from s , taking action a , and thereafter following policy π is defined by

$$Q^\pi(s, a) = E_\pi\{R_t | s_t = s, a_t = a\} = E_\pi\left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t = s, a_t = a\right\}.$$

A value function that returns the highest value for the best action in each state is known as the *optimal value function*. $V^*(s)$ and $Q^*(s, a)$ denote the optimal state and action value functions and are given respectively by

$$V^*(s) = \max_a \sum_{s'} P_{ss'}^a [R_{ss'}^a + \gamma V^*(s')],$$

$$Q^*(s, a) = \sum_{s'} P_{ss'}^a [R_{ss'}^a + \gamma \max_{a'} Q^*(s', a')].$$

Learning algorithms are concerned with how and when to update the value function using provided rewards. The differences in algorithms range depending on the required data that they need to operate, how they perform calculations and finally when this update takes place. Learning algorithms can be divided into three major classes: *dynamic programming*, *Monte-Carlo method* and *time-difference method*.

Dynamic programming (DP) works by assigning blame to the many decisions a system has to do while operating, this is done using two simple principles. Firstly, if an action causes something bad to happen immediately, then it learns not to do that action from that state again. Secondly, if all actions from a certain state lead to a bad result then that state should also be avoided. DP requires a *perfect environment model* in order to find a solution. Therefore the environment must have finite sets of states S and actions $A(s)$, and also finite sets of transition probabilities $P_{ss'}^a = \Pr\{s_{t+1} = s' | s_t = s, a_t = a\}$ and immediate rewards $R_{ss'}^a = E\{r_{t+1} | s_{t+1} = s', s_t = s, a_t = a\}$ for all $s \in S, a \in A(s)$. The value function in DP is updated using the equation

$$V^\pi(s) = \sum_a \pi(s, a) \sum_{s'} P_{ss'}^a [R_{ss'}^a + \gamma V^*(s')].$$

Starting from the far right in this equation it can be seen that the reward received for taking an action is added to the discounted value of the resulting state of that action. However, a single action may have multiple effects in a complex environment leading to multiple resulting states. The value of each possible resulting state is multiplied by the corresponding transition probability and all results are added to get the actual value of a single action. In order to calculate the value of the state itself, the value of each action is calculated and added to produce the full value of the state.

The two biggest problems encountered when developing applications using DP are [Sio05]: (i) the requirement of previously knowing all effects of actions taken in the environment, and (ii) the exponential increase in computation required to calculate the value of a state for only a small increase in possible actions and/or effects.

Monte Carlo (MC) methods however do not assume complete knowledge of the environment and require only experience through sampling sequences of states, actions and rewards from direct interaction with an environment. They are able to learn by segmenting sequences of actions into episodes and averaging rewards received as shown by the following algorithm [SB98]:

```

1:  $\pi \leftarrow$  policy to be evaluated;
2:  $V \leftarrow$  an arbitrary state-value function;
3:  $Returns(s) \leftarrow$  an empty list, for all  $s \in S$ ;
4: while true do;
5:   Generate an episode using;
6:   for each state  $s$  appearing in the episode do;
7:      $R \leftarrow$  return following the first occurrence of  $s$ ;
8:     Append  $R$  to  $Returns(s)$ ;
9:      $V(s) \leftarrow average(Returns(s))$ ;
10:  end for;
11: end while;
```

Note that the algorithm requires the generation of an entire episode (line 5) before performing any updates to the value function.

MC is also able to estimate action values rather than state values, in this case policy evaluation is performed by estimating $Q^\pi(s, a)$, which is the expected return when starting in state s , taking action a , and thereafter following policy π . The relevant algorithm has the same structure as above. When MC is used for approximating optimal policies, the *generalized policy iteration* (GPI) is used. GPI maintains an approximate policy and an approximate value function, it then performs policy evaluation²⁰ and policy improvement²¹ repeatedly. This means that the value function is updated to reflect the current policy while the policy is then improved with respect to the value function. Using these two processes GPI is able to maximize its rewards.

Temporal-Difference (TD) learning combines ideas from both MC and DP methods. Similarly to MC, TD methods are able to learn from experiences and do not need a model of the environment's dynamics. Like DP, TD methods update the value function based in part on estimates of future states (this feature is called *bootstrapping*) and hence do not require waiting for the episode to finish. An example of TD learning is the *Sarsa* algorithm [SB98]:

```

1: Initialize  $Q(s, a)$  arbitrarily;
2: for each episode do;
3:   Initialize  $s$ ;
4:   Choose  $a$  from  $s$  using policy derived from  $Q$ ;
5:   for each state  $s$  in episode do;
```

²⁰ Policy evaluation calculates the value function of a given policy.

²¹ Policy improvement changes the policy such that it takes the best actions as dictated by the value function.

```

6:   Take action  $a$ , observe  $r, s'$ ;
7:   Choose  $a'$  from  $s'$  using policy derived from  $Q$ ;
8:    $Q(s, a) \leftarrow Q(s, a) + \alpha[r + \gamma Q(s', a') - Q(s, a)]$ ;
9:    $s \leftarrow s'; a \leftarrow a'$ ;
10: end for;
11: end for;

```

The most important part of the algorithm is line 8 where the action value function is updated according to the rule:

$$Q(s, a) \leftarrow Q(s, a) + \alpha[r + \gamma Q(s', a') - Q(s, a)],$$

where α is called the step-size parameter and it controls how much the value function is changed with each update. Sarsa is an on-policy TD-algorithm and it requires the agent to select the following action before updating $Q(s, a)$. This is because $Q(s, a)$ is calculated by subtracting $Q(s, a)$ from the discounted value of $Q(s', a')$, which can only be known by selecting a' . Note that actions are selected using a policy that is based on the value function and in turn the value function is updated from the reward received.

Off-policy TD is able to approximate the optimal value function independently of the policy being followed. An example is the *Qlearning* algorithm [SB98]:

```

1: Initialize  $Q(s, a)$  arbitrarily;
2: for each episode do;
3:   Initialize  $s$ ;
4:   for Each state  $s$  in episode do;
5:     Choose  $a'$  from  $s'$  using policy derived from  $Q$ ;
6:     Take action  $a$ , observe  $r, s'$ ;
7:      $Q(s, a) \leftarrow Q(s, a) + \alpha[r + \gamma \max_{a'} Q(s', a') - Q(s, a)]$ ;
8:      $s \leftarrow s'$ ;
9:   end for;
10: end for;

```

The main difference between Sarsa and Qlearning lies in the calculation that updates the value function, the Qlearning update function is given by

$$Q(s, a) \leftarrow Q(s, a) + \alpha[r + \gamma \max_{a'} Q(s', a') - Q(s, a)].$$

With Sarsa the value function is updated based on the *next chosen action*, while with Qlearning it is updated based on the *best known future action* even if that action is actually *not selected* in the next iteration of the algorithm.

Exploration versus exploitation. One of the more well known problems within the RL literature is the *exploration/exploitation problem*. During its operation the agent forms the *action estimates* $Q^\pi(a) = Q^*(a)$. The best known action at time t would therefore be

$$a_t^* = \arg \max_a Q_t(a).$$

An agent is said to be *exploring* when it tries an new action for a particular situation $a \neq a_t^*$. The reward obtained from the execution of that action is used to update the value function accordingly. An agent is said to be *exploiting* its learning knowledge when it chooses the *greedy action* (i.e., best action) indicated by its value function in a particular state $a = a_t^*$. In this case, the agent also updates the value function according to the reward received. This may have two effects, firstly, the reward may be similar to the one expected by the value function, which means that the value function is stabilizing on the problem trying to be solved. Secondly, it may be totally different to the value expected, therefore changing the value function and possibly the ordering of the actions with respect the their values. Hence, another action may subsequently become the ‘best’ action for that state.

An action selection policy controls the exploitation/exploration that is performed by the agent while learning. There are two types of policies commonly considered. Firstly, the *EGreedy policy* explores by selecting actions randomly but only for a defined percentage of all actions chosen as

$$a_t = \begin{cases} a_t^* & \text{if } PR = (1 - \epsilon), \\ \text{random} & \text{if } PR = \epsilon. \end{cases}$$

For example, if $\epsilon = 0.1$ then the agent will explore only 10% of the time, the rest of the time it chooses the greedy action.

Secondly, the *SoftMax action selection* is more complex. It makes its choice based on the relation

$$a_t = \frac{e^{Q_t(a)/\tau}}{\sum_{b=1}^n e^{Q_t(b)/\tau}},$$

where τ is called the *temperature value*. A high temperature selects all actions randomly, while a low temperature selects actions in a greedy fashion. An intermediate temperature value causes SoftMax to select actions with a probability that is based on their value. This way actions with a high value have a greater chance of being selected while actions with a lower value have less chance of being selected. The advantage of SoftMax is that it tends to select the best action most of the time followed by the second-best, the third-best and so on, an action with a very low value is seldom executed. This is useful when a particular action is known to cause extremely bad rewards. Using SoftMax, that action will always get a very small probability of execution, with EGreedy however, it has the same probability as any other action when exploring.

2.4 Neural–Networks Approach

In this section we give a brief review on neural networks approach to brain and cognition. Formally, here we are dealing with a *network evolution* 2–functor $\mathcal{E}$, given by

$$\begin{array}{ccc}
 \begin{array}{ccc}
 A & \xrightarrow{f} & B \\
 \downarrow h & \text{CURRENT} & \downarrow g \\
 & \text{NET} & \\
 C & \xrightarrow{k} & D \\
 & \text{STATE} &
 \end{array}
 & \xrightarrow{\mathcal{E}} &
 \begin{array}{ccc}
 \mathcal{E}(A) & \xrightarrow{\mathcal{E}(f)} & \mathcal{E}(B) \\
 \downarrow \mathcal{E}(h) & \text{DESIRED} & \downarrow \mathcal{E}(g) \\
 & \text{NET} & \\
 \mathcal{E}(C) & \xrightarrow{\mathcal{E}(k)} & \mathcal{E}(D) \\
 & \text{STATE} &
 \end{array}
 \end{array}$$

Here $\mathcal{E}$ represents an association/projection functor from the *source* 2–category of the current network state, defined as a commutative square of small network categories $A, B, C, D, \dots$ of *current network components* and their causal interrelations $f, g, h, k, \dots$, onto the *target* 2–category of the desired network state, defined as a commutative square of small network categories $\mathcal{E}(A), \mathcal{E}(B), \mathcal{E}(C), \mathcal{E}(D), \dots$ of *evolved network components* and their causal interrelations $\mathcal{E}(f), \mathcal{E}(g), \mathcal{E}(h), \mathcal{E}(k)$. As in the previous section, each causal arrow in above diagram, e.g., $f : A \rightarrow B$, stands for a *generic network dynamorphism*.

Recall that an *artificial neural network* (ANN), also called a *simulated neural network* (SNN) or commonly just *neural network* (NN) is an interconnected group of artificial neurons that uses a mathematical or computational model for information processing based on the so–called *connectionist approach* to computation. In most cases an ANN is an *adaptive system* that changes its structure based on external or internal information that flows through the network.

In more practical terms neural networks are nonlinear statistical data modeling tools. They can be used to model complex relationships between inputs and outputs or to find patterns in data.

Dynamically, the ANNs are *nonlinear dynamical systems* that act as *functional approximators* [Kos92]. The ANN builds *discriminant functions* from its processing elements (PE)s. The ANN topology determines the *number* and *shape* of the discriminant functions. The shapes of the discriminant functions change with the topology, so ANNs are considered *semi–parametric classifiers*. One of the central advantages of ANNs is that they are sufficiently powerful to create arbitrary discriminant functions so ANNs can achieve optimal classification.

The placement of the discriminant functions is controlled by the network weights. Following the ideas of non–parametric training, the weights are adjusted directly from the training data without any assumptions about the data’s statistical distribution. Hence one of the central issues in neural network design is to utilize systematic procedures, the so–called *training algo-*

rithm, to modify the weights so that as accurate a classification as possible is achieved. The accuracy is quantified by an error criterion [PEL00].

The training is usually performed in the following way. First, data is presented, and an output is computed. An error is obtained by comparing the output $\{y\}$ with a desired response $\{d\}$ and it is used to modify the weights with a training algorithm. This procedure is repeated using all the data in the training set until a convergence criterion is met. Thus, in ANNs (and in adaptive systems in general) the designer does not have to specify the *parameters* of the system. They are automatically extracted from the input data and the desired response by means of the training algorithm. The two central issues in neural network design (semi-parametric classifiers) are the selection of the shape and number of the discriminant functions and their placement in pattern space such that the classification error is minimized [PEL00].

2.4.1 Biological Versus Artificial Neural Nets

In biological neural networks, signals are transmitted between neurons by electrical pulses (action potentials or spike trains) traveling along the axon. These pulses impinge on the afferent neuron at terminals called synapses. These are found principally on a set of branching processes emerging from the cell body (soma) known as dendrites. Each pulse occurring at a synapse initiates the release of a small amount of chemical substance or neurotransmitter which travels across the synaptic cleft and which is then received at postsynaptic receptor sites on the dendritic side of the synapse. The neurotransmitter becomes bound to molecular sites here which, in turn, initiates a change in the dendritic membrane potential. This postsynaptic potential (PSP) change may serve to increase (hyperpolarize) or decrease (depolarize) the polarization of the postsynaptic membrane. In the former case, the PSP tends to inhibit generation of pulses in the afferent neuron, while in the latter, it tends to excite the generation of pulses. The size and type of PSP produced will depend on factors such as the geometry of the synapse and the type of neurotransmitter. Each PSP will travel along its dendrite and spread over the soma, eventually reaching the base of the axon (axonhillock). The afferent neuron sums or integrates the effects of thousands of such PSPs over its dendritic tree and over time. If the integrated potential at the axonhillock exceeds a threshold, the cell fires and generates an action potential or spike which starts to travel along its axon. This then initiates the whole sequence of events again in neurons contained in the efferent pathway.

ANNs are very loosely based on these ideas. In the most general terms, a ANN consists of large numbers of simple processors linked by weighted connections. By analogy, the processing nodes may be called artificial neurons. Each node output depends only on information that is locally available at the node, either stored internally or arriving via the weighted connections. Each unit receives inputs from many other nodes and transmits its output to yet other nodes. By itself, a single processing element is not very powerful; it

generates a scalar output, a single numerical value, which is a simple nonlinear function of its inputs. The power of the system emerges from the combination of many units in an appropriate way [FS92].

ANN is specialized to implement different functions by varying the connection topology and the values of the connecting weights. Complex functions can be implemented by connecting units together with appropriate weights. In fact, it has been shown that a sufficiently large network with an appropriate structure and properly chosen weights can approximate with arbitrary accuracy any function satisfying certain broad constraints. In ANNs, the design motivation is what distinguishes them from other mathematical techniques: an ANN is a processing device, either an algorithm, or actual hardware, whose design was motivated by the design and functioning of animal brains and components thereof.

There are many different types of ANNs, each of which has different strengths particular to their applications. The abilities of different networks can be related to their structure, dynamics and learning methods.

2.4.2 Common Discrete ANNs

Multilayer Perceptrons

The most common ANN model is the *feedforward neural network* with one input layer, one output layer, and one or more hidden layers, called *multilayer perceptron* (MLP, see Figure 2.38). This type of neural network is known as a *supervised network* because it requires a desired output in order to learn. The goal of this type of network is to *create a model* $f : x \rightarrow y$ that correctly maps the input x to the output y using historical data so that the model can then be used to produce the output when the desired output is unknown [Kos92].

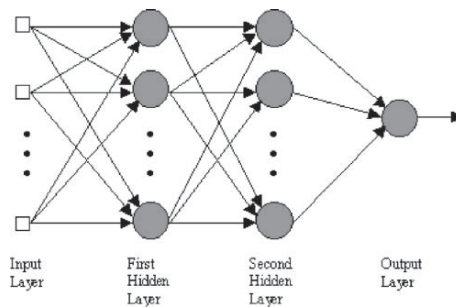


Fig. 2.38. Multilayer perceptron (MLP) with two hidden layers.

In MLP the inputs are fed into the input layer and get multiplied by interconnection weights as they are passed from the input layer to the first hidden layer. Within the first hidden layer, they get summed then processed

by a nonlinear function (usually the hyperbolic tangent). As the processed data leaves the first hidden layer, again it gets multiplied by interconnection weights, then summed and processed by the second hidden layer. Finally the data is multiplied by interconnection weights then processed one last time within the output layer to produce the neural network output.

MLPs are typically trained with *static backpropagation*. These networks have found their way into countless applications requiring static pattern classification. Their main advantage is that they are easy to use, and that they can approximate any input/output map. The key disadvantages are that they train slowly, and require lots of training data (typically three times more training samples than the number of network weights).

McCulloch–Pitts Processing Element

MLPs are typically composed of *McCulloch–Pitts neurons* (see [MP43]). This processing element (PE) is simply a sum-of-products followed by a threshold nonlinearity. Its input–output equation is

$$y = f(\text{net}) = f(w_i x^i + b), \quad (i = 1, \dots, D),$$

where D is the number of inputs, x^i are the inputs to the PE, w_i are the weights and b is a bias term (see e.g., [MP69]). The activation function is a *hard threshold* defined by *signum* function,

$$f(\text{net}) = \begin{cases} 1, & \text{for } \text{net} \geq 0, \\ -1, & \text{for } \text{net} < 0. \end{cases}$$

Therefore, McCulloch–Pitts PE is composed of an adaptive linear element (*Adaline*, the weighted sum of inputs), followed by a signum nonlinearity [PEL00].

Sigmoidal Nonlinearities

Besides the hard threshold defined by signum function, other nonlinearities can be utilized in conjunction with the McCulloch–Pitts PE. Let us now smooth out the threshold, yielding a sigmoid shape for the nonlinearity. The most common nonlinearities are the *logistic* and the *hyperbolic tangent threshold activation functions*,

$$\begin{aligned} \text{hyperbolic} : \quad & f(\text{net}) = \tanh(\alpha \text{net}), \\ \text{logistic} : \quad & f(\text{net}) = \frac{1}{1 + \exp(-\alpha \text{net})}, \end{aligned}$$

where α is a *slope parameter* and normally is set to 1. The major difference between the two sigmoidal nonlinearities is the range of their output values. The logistic function produces values in the interval $[0, 1]$, while the hyperbolic tangent produces values in the interval $[-1, 1]$. An alternate interpretation

of this PE substitution is to think that the discriminant function has been generalized to

$$g(x) = f(w_i x^i + b), \quad (i = 1, \dots, D),$$

which is sometimes called a *ridge* function. The combination of the synapse and the tanh axon (or the sigmoid axon) is usually referred to as the modified McCulloch–Pitts PE, because they all respond to the full input space in basically the same functional form (a sum of products followed by a global nonlinearity). The output of the logistic function varies from 0 to 1. Under some conditions, the logistic function allows a very powerful interpretation of the output of the PE as a posteriori probabilities for Gaussian-distributed input classes. The tanh is closely related to the logistic function by a linear transformation in the input and output spaces, so neural networks that use either of these can be made equivalent by changing weights and biases [PEL00].

Gradient Descent on the Net's Performance Surface

The *search* for the weights to meet a *desired response* or internal constraint is the essence of any *connectionist* computation. The central problem to be solved on the road to machine-based classifiers is how to automate the process of *minimizing the error* so that the machine can independently make these weight changes, without need for hidden agents, or external observers. The optimality criterion to be minimized is usually the *mean square error* (MSE)

$$J = \frac{1}{2N} \sum_{i=1}^N \varepsilon_i^2,$$

where ε_i is the instantaneous error that is added to the output y_i (the linearly fitted value), and N is the number of observations. The function $J(w)$ is called the *performance surface* (the total error surface plotted in the space of weights w).

The search for the minimum of a function can be done efficiently using a broad class of methods based on *gradient information*. The gradient has two main advantages for the search:

1. It can be computed locally, and
2. It always points in the direction of maximum change.

The *gradient of the performance surface*, $\nabla J = \nabla_w J$, is a vector (with the dimension of w) that always points toward the direction of maximum J -change and with a magnitude equal to the slope of the tangent of the performance surface. The minimum value of the error J_{min} depends on both the input signal x^i and the desired signal d_i ,

$$J_{min} = \frac{1}{2N} \left[\sum_i d_i^2 - \frac{(d_i x^i)}{\sum_i x^i} \right], \quad (i = 1, \dots, D).$$

The location in coefficient space where the minimum w^* occurs also depends on both x^i and d_i . The performance surface shape depends only on the input signal x^i [PEL00].

Now, if the goal is to reach the minimum, the search must be in the direction opposite to the gradient. The overall method of gradient searching can be stated in the following way: Start the search with an arbitrary initial weight $w(0)$, where the iteration number is denoted by the index in parentheses. Then compute the gradient of the performance surface at $w(0)$, and modify the initial weight proportionally to the negative of the gradient at $w(0)$. This changes the operating point to $w(1)$. Then compute the gradient at the new position $w(1)$, and apply the same procedure again, that is,

$$w(n+1) = w(n) - \eta \nabla J(n),$$

where η is a small constant and $\nabla J(n)$ denotes the gradient of the performance surface at the n th iteration. The constant η is used to maintain stability in the search by ensuring that the operating point does not move too far along the performance surface. This search procedure is called the *steepest descent method*.

In the late 1960s, Widrow proposed an extremely elegant algorithm to estimate the gradient that revolutionized the application of gradient descent procedures. His idea is very simple: Use the instantaneous value as the estimator for the true quantity:

$$\nabla J(n) = \frac{\partial}{\partial w(n)} J \approx \frac{1}{2} \frac{\partial}{\partial w(n)} (\varepsilon^2(n)) = -\varepsilon(n) x(n),$$

i.e., instantaneous estimate of the gradient at iteration n is simply the product of the current input $x(n)$ to the weight $w(n)$ times the current error $\varepsilon(n)$. The amazing thing is that the gradient can be estimated with one multiplication per weight. This is the gradient estimate that led to the celebrated *least means square algorithm* (LMS):

$$w(n+1) = w(n) + \eta \varepsilon(n) x(n), \quad (2.111)$$

where the small constant η is called the *step size*, or the *learning rate*. The estimate will be noisy, however, since the algorithm uses the error from a single sample instead of summing the error for each point in the data set (e.g., the MSE is estimated by the error for the current sample).

Now, for fast convergence to the neighborhood of the minimum a large step size is desired. However, the solution with a large step size suffers from rattling. One attractive solution is to use a large learning rate in the beginning of training to move quickly toward the location of the optimal weights, but then the learning rate should be decreased to get good accuracy on the final weight values. This is called *learning rate scheduling*. This simple idea can be implemented with a variable step size controlled by

$$\eta(n+1) = \eta(n) - \beta,$$

where $\eta(0) = \eta_0$ is the initial step size, and β is a small constant [PEL00].

Perceptron and Its Learning Algorithm

Rosenblatt perceptron (see [Ros58b, MP69]) is a *pattern-recognition* machine that was invented in the 1950s for optical character recognition. The perceptron has an input layer fully connected to an output layer with multiple McCulloch–Pitts PEs,

$$y_i = f(\text{net}_i) = f(w_i x^i + b_i), \quad (i = 1, \dots, D),$$

where b_i is the bias for each PE. The number of outputs y_i is normally determined by the number of classes in the data. These PEs add the individual scaled contributions and respond to the entire input space.

F. Rosenblatt proposed the following procedure to directly minimize the error by changing the weights of the McCulloch–Pitts PE: Apply an input example to the network. If the output is correct do nothing. If the response is incorrect, tweak the weights and bias until the response becomes correct. Get the next example and repeat the procedure, until all the patterns are correctly classified. This procedure is called the *perceptron learning algorithm*, which can be put into the following form:

$$w(n+1) = w(n) + \eta(d(n) - y(n))x(n),$$

where η is the step size, y is the network output, and d is the desired response.

Clearly, the functional form is the same as in the LMS algorithm (2.111), that is, the old weights are incrementally modified proportionally to the product of the error and the input, but there is a significant difference. We cannot say that this corresponds to gradient descent since the system has a discontinuous nonlinearity. In the perceptron learning algorithm, $y(n)$ is the output of the nonlinear system. The algorithm is directly minimizing the difference between the response of the McCulloch–Pitts PE and the desired response, instead of minimizing the difference between the Adaline output and the desired response [PEL00].

This subtle modification has tremendous impact on the performance of the system. For one thing, the McCulloch–Pitts PE learns only when its output is wrong. In fact, when $y(n) = d(n)$, the weights remain the same. The net effect is that the final values of the weights are no longer equal to the linear regression result, because the nonlinearity is brought into the weight update rule. Another way of phrasing this is to say that the weight update became much more selective, effectively gated by the system performance. Notice that the LMS update is also a function of the error to a certain degree. Larger errors have more effect on the weight update than small errors, but all patterns affect the final weights implementing a ‘smooth gate’. In the perceptron the net effect is that the placement of the discriminant function is no longer controlled smoothly by all the input samples as in the Adaline, only by the ones that are important for placing the discriminant function in a way that explicitly minimizes the output error.

The Delta Learning Rule

One can show that the LMS rule is equivalent to the chain rule in the computation of the *sensitivity* of the cost function J with respect to the unknowns. Interpreting the LMS equation (2.111) with respect to the sensitivity concept, we see that the gradient measures the sensitivity. LMS is therefore updating the weights proportionally to how much they affect the performance, i.e., proportionally to their sensitivity.

The LMS concept can be extended to the McCulloch–Pitts PE, which is a nonlinear system. The main question here is how can we compute the sensitivity through a nonlinearity? [PEL00] The so-called δ -rule represents a direct extension of the LMS rule to nonlinear systems with smooth nonlinearities. In case of the McCulloch–Pitts PE, *delta-rule* reads:

$$w_i(n+1) = w_i(n) + \eta \varepsilon_p(n) x_p^i(n) f'_p(\text{net}(n)),$$

where $f'(\text{net})$ is the partial derivative of the static nonlinearity, such that the *chain rule* is applied to the network topology, i.e.,

$$f'(\text{net}) x^i = \frac{\partial y}{\partial w_i} = \frac{\partial y}{\partial \text{net}} \frac{\partial}{\partial w_i}. \quad (2.112)$$

As long as the PE nonlinearity is smooth we can compute how much a change in the weight δw_i affects the output y , or from the point of view of the sensitivity, how sensitive the output y is to a change in a particular weight δw_i . Note that we compute this output sensitivity by a product of partial derivatives through intermediate points in the topology. For the nonlinear PE there is only one intermediate point, *net*, but we really do not care how many of these intermediate points there are. The chain rule can be applied as many times as necessary. In practice, we have an error at the output (the difference between the desired response and the actual output), and we want to adjust all the PE weights so that the error is minimized in a statistical sense. The obvious idea is to distribute the adjustments according to the sensitivity of the output to each weight.

To modify the weight, we actually *propagate back the output error* to intermediate points in the network topology and scale it along the way as prescribed by (2.112) according to the element transfer functions:

$$\begin{aligned} \text{forward path : } & x^i \mapsto w_i \mapsto \text{net} \mapsto y \\ \text{backward path 1 : } & w_i \xleftarrow{\partial \text{net} / \partial w} \text{net} \xleftarrow{\partial y / \partial \text{net}} y \\ \text{backward path 2 : } & w_i \xleftarrow{\partial y / \partial w} y. \end{aligned}$$

This methodology is very powerful, because we do not need to know explicitly the error at intermediate places, such as *net*. The chain rule automatically derives the error contribution for us. This observation is going to be crucial for

adapting more complicated topologies and will result in the *backpropagation* algorithm, discovered in 1988 by Werbos [Wer89].

Now, several key aspects have changed in the performance surface (which describes how the cost changes with the weights) with the introduction of the nonlinearity. The nice, parabolic performance surface of the linear least squares problem is lost. The performance depends on the topology of the network through the output error, so when nonlinear processing elements are used to solve a given problem the ‘performance – weights’ relationship becomes nonlinear, and there is no guarantee of a single minimum. The performance surface may have several minima. The minimum that produces the smallest error in the search space is called the *global minimum*. The others are called *local* minima. Alternatively, we say that the performance surface is *nonconvex*. This affects the search scheme because gradient descent uses local information to search the performance surface. In the immediate neighborhood, local minima are indistinguishable from the global minimum, so the gradient search algorithm may be caught in these suboptimal performance points, ‘thinking’ it has reached the global minimum [PEL00].

δ –rule extended to perceptron reads:

$$w_{ij}(n+1) = w_{ij}(n) - \eta \frac{\partial J}{\partial w_{ij}} = w_{ij}(n) + \eta \delta_{ip} x_p^j,$$

which are local quantities available at the weight, that is, the activation x_p^j that reaches the weight w_{ij} from the input and the local error δ_{ip} propagated from the cost function J . This algorithm is local to the weight. Only the local error δ_i and the local activation x^j are needed to update a particular weight. This means that it is immaterial how many PEs the net has and how complex their interconnection is. The training algorithm can concentrate on each PE individually and work only with the local error and local activation [PEL00].

Backpropagation

The multilayer perceptron constructs input–output mappings that are a nested composition of nonlinearities, that is, they are of the form

$$y = f \left(\sum f \left(\sum (\cdot) \right) \right),$$

where the number of function compositions is given by the number of network layers. The resulting map is very flexible and powerful, but it is also hard to analyze [PEL00].

MLPs are usually trained by generalized δ –rule, the so-called *backpropagation* (BP). The weight update using backpropagation is

$$w_{ij}(n+1) = w_{ij}(n) + \eta f'(\text{net}_i(n)) \left(\varepsilon^k(n) f'(\text{net}_k(n)) w_{ki}(n) \right) y_j(n). \quad (2.113)$$

The summation in (2.113) is a sum of local errors δ_k at each network output PE, scaled by the weights connecting the output PEs to the i th PE. Thus the term in parenthesis in (2.113) effectively computes the total error reaching the i th PE from the output layer (which can be thought of as the i th PE's contribution to the output error). When we pass it through the i th PE nonlinearity, we have its local error, which can be written as

$$\delta_i(n) = f'_i(\text{net}(n)) \delta^k w_{ki}(n).$$

Thus there is a unifying link in all the gradient-descent algorithms. All the weights in gradient descent learning are updated by multiplying the local error $\delta_i(n)$ by the local activation $x^j(n)$ according to Widrow's estimation of the instantaneous gradient first shown in the LMS rule:

$$\Delta w_{ij}(n) = \eta \delta_i(n) y_j(n).$$

What differs is the calculation of the local error, depending on whether the PE is linear or nonlinear and if the weight is attached to an output PE or a hidden-layer PE [PEL00].

Momentum Learning

Momentum learning is an improvement to the straight gradient-descent search in the sense that a memory term (the past increment to the weight) is used to speed up and stabilize convergence. In *momentum learning* the equation to update the weights becomes

$$w_{ij}(n+1) = w_{ij}(n) + \eta \delta_i(n) x_j(n) + \alpha (w_{ij}(n) - w_{ij}(n-1)),$$

where α is the momentum constant, usually set between 0.5 and 0.9. This is called momentum learning due to the form of the last term, which resembles the momentum in mechanics. Note that the weights are changed proportionally to how much they were updated in the last iteration. Thus if the search is going down the hill and finds a flat region, the weights are still changed, not because of the gradient (which is practically zero in a flat spot), but because of the rate of change in the weights. Likewise, in a narrow valley, where the gradient tends to bounce back and forth between hillsides, the momentum stabilizes the search because it tends to make the weights follow a smoother path. Imagine a ball (weight vector position) rolling down a hill (performance surface). If the ball reaches a small flat part of the hill, it will continue past this local minimum because of its momentum. A ball without momentum, however, will get stuck in this valley. Momentum learning is a robust method to speed up learning, and is usually recommended as the default search rule for networks with nonlinearities.

Advanced Search Methods

The popularity of *gradient descent method* is based more on its simplicity (it can be computed locally with two multiplications and one addition per weight) than on its search power. There are many other search procedures more powerful than backpropagation. For example, *Newtonian method* is a second-order method because it uses the information on the curvature to adapt the weights. However Newtonian method is computationally much more costly to implement and requires information not available at the PE, so it has been used little in neurocomputing. Although more powerful, Newtonian method is still a local search method and so may be caught in local minima or diverge due to the difficult neural network performance landscapes. Other techniques such as *simulated annealing*²² and *genetic algorithms* (GA)²³ are global search procedures, that is, they can avoid local minima. The issue is that they are more costly to implement in a distributed system like a neural network, either because they are inherently slow or because they require nonlocal quantities [PEL00].

The problem of search with local information can be formulated as an approximation to the functional form of the *matrix cost function* $J(\mathbf{w})$ at the operating point $\mathbf{w}_0$. This immediately points to the Taylor series expansion of J around $\mathbf{w}_0$,

$$J(\mathbf{w} - \mathbf{w}_0) = J_0 + (\mathbf{w} - \mathbf{w}_0)\nabla J_0 + \frac{1}{2}(\mathbf{w} - \mathbf{w}_0)\mathbf{H}_0(\mathbf{w} - \mathbf{w}_0)^T + \dots,$$

where ∇J is our familiar gradient, and $\mathbf{H}$ is the Hessian matrix, that is, the matrix of second derivatives with entries

$$H_{ij}(\mathbf{w}_0) = \left. \frac{\partial^2 J(w)}{\partial w_i \partial w_j} \right|_{w=\mathbf{w}_0},$$

evaluated at the operating point. We can immediately see that the Hessian cannot be computed with the information available at a given PE, since it uses information from two different weights. If we differentiate J with respect to the weights, we get

$$\nabla J(\mathbf{w}) = \nabla J_0 + \mathbf{H}_0(\mathbf{w} - \mathbf{w}_0) + \dots \quad (2.114)$$

²² Simulated annealing is a global search criterion by which the space is searched with a random rule. In the beginning the variance of the random jumps is very large. Every so often the variance is decreased, and a more local search is undertaken. It has been shown that if the decrease of the variance is set appropriately, the global optimum can be found with probability one. The method is called simulated annealing because it is similar to the annealing process of creating crystals from a hot liquid.

²³ Genetic algorithms are global search procedures proposed by J. Holland that search the performance surface, concentrating on the areas that provide better solutions. They use 'generations' of search points computed from the previous search points using the operators of crossover and mutation (hence the name).

so we can see that to compute the full gradient at $\mathbf{w}$ we need all the higher terms of the derivatives of J . This is impossible. Since the performance surface tends to be bowl shaped (quadratic) near the minimum, we are normally interested only in the first and second terms of the expansion [PEL00].

If the expansion of (2.114) is restricted to the first term, we get the gradient-search methods (hence they are called *first-order-search methods*), where the gradient is estimated with its value at $\mathbf{w}_0$. If we expand to use the second-order term, we get *Newton method* (hence the name second-order method). If we equate the truncated relation (2.114) to 0 we immediately get

$$w = w_0 - \mathbf{H}_0^{-1} \nabla J_0,$$

which is the equation for the Newton method, which has the nice property of quadratic termination (it is guaranteed to find the exact minimum in a finite number of steps for quadratic performance surfaces). For most quadratic performance surfaces it can converge in one iteration.

The real difficulty is the memory and the computational cost (and precision) to estimate the Hessian. Neural networks can have thousands of weights, which means that the Hessian will have millions of entries. This is why methods of approximating the Hessian have been extensively researched. There are two basic classes of approximations [PEL00]:

1. Line search methods, and
2. Pseudo-Newton methods.

The information in the first type is restricted to the gradient, together with line searches along certain directions, while the second seeks approximations to the Hessian matrix. Among the line search methods probably the most effective is the *conjugate gradient method*. For quadratic performance surfaces the conjugate gradient algorithm preserves quadratic termination and can reach the minimum in D steps, where D is the dimension of the weight space. Among the Pseudo-Newton methods probably the most effective is the *Levenberg-Marquardt algorithm* (LM), which uses the Gauss-Newton method to approximate the Hessian. LM is the most interesting for neural networks, since it is formulated as a sum of quadratic terms just like the cost functions in neural networks.

The *extended Kalman filter* (EKF, see (2.2.1) above) forms the basis of a second-order neural network training method that is a practical and effective alternative to the batch-oriented, second-order methods mentioned above. The essence of the recursive EKF procedure is that, during training, in addition to evolving the weights of a network architecture in a sequential (as opposed to batch) fashion, an approximate error covariance matrix that encodes second-order information about the training problem is also maintained and evolved.

Homotopy Methods

The most popular method for solving nonlinear equations in general is the *Newton-Raphson method*. Unfortunately, this method sometimes fails, especially in cases when nonlinear equations possess multiple solutions (zeros). An emerging family of methods that can be used in such cases are homotopy (continuation) methods (see chapter 2). These methods are robust and have good convergence properties.

Homotopy methods or *continuation methods* have increasingly been used for solving variety of nonlinear problems in fluid dynamics, structural mechanics, systems identifications, and integrated circuits (see [Wat90]). These methods, popular in mathematical programming, are globally convergent provided that certain coercivity and continuity conditions are satisfied by the equations that need to be solved [Wat90]. Moreover, they often yield all the solutions to the nonlinear system of equations.

The idea behind a homotopy or continuation method is to embed a parameter λ in the nonlinear equations to be solved. This is why they are sometimes referred to as *embedding methods*. Initially, parameter λ is set to zero, in which case the problem is reduced to an easy problem with a known or easily-found solution. The set of equations is then gradually deformed into the originally posed difficult problem by varying the parameter λ . The original problem is obtained for $\lambda = 1$. Homotopies are a class of continuation methods, in which parameter λ is a function of a path arc length and may actually increase or decrease as the path is traversed. Provided that certain coercivity conditions imposed on the nonlinear function to be solved are satisfied, the homotopy path does not branch (bifurcate) and passes through all the solutions of the nonlinear equations to be solved.

The zero curve of the homotopy map can be tracked by various techniques: an *ODE-algorithm*, a *normal flow algorithm*, and an *augmented Jacobian matrix algorithm*, among others [Wat90].

As a typical example, homotopy techniques can be applied to find the zeros of the gradient function $F : \mathbb{R}^N \rightarrow \mathbb{R}^N$, such that

$$F(\theta) = \frac{\partial E(\theta)}{\partial \theta_k}, \quad 1 \leq k \leq N,$$

where $E = E(\theta)$ is the certain error function dependent on N parameters θ_k . In other words, we need to solve a system of nonlinear equations

$$F(\theta) = 0. \quad (2.115)$$

In order to solve equation (2.115), we can create a linear homotopy function

$$H(\theta, \lambda) = (1 - \lambda)(\theta - a) + \lambda F(\theta),$$

where a is an arbitrary starting point. Function $H(\theta, \lambda)$ has properties that equation $H(\theta, 0) = 0$ is easy to solve, and that $H(\theta, 1) \equiv F(\theta)$.

ANNs as Functional Approximators

The *universal approximation theorem* of Kolmogorov states [Hay94]:

Let $\phi(\cdot)$ be a nonconstant, bounded, and monotone-increasing continuous (C^0) function. Let I^N denote N D unit hypercube $[0, 1]^N$. The space of C^0 -functions on I^N is denoted by $C(I^N)$. Then, given any function $f \in C(I^N)$ and $\epsilon > 0$, there exist an integer M and sets of real constants $\alpha_i, \theta_i, \omega_{ij}, i = 1, \dots, M; j = 1, \dots, N$ such that we may define

$$F(x_1, \dots, x_N) = \alpha_i \phi(\omega_{ij} x_j - \theta_i),$$

as an approximate realization of the function $f(\cdot)$; that is

$$|F(x_1, \dots, x_N) - f(x_1, \dots, x_N)| < \epsilon \quad \text{for all } \{x_1, \dots, x_N\} \in I^N.$$

This theorem is directly applicable to *multilayer perceptrons*. First, the logistic function $1/[1 + \exp(-v)]$ used as the sigmoidal nonlinearity in a neuron model for the construction of a multilayer perceptron is indeed a nonconstant, bounded, and monotone-increasing function; it therefore satisfies the conditions imposed on the function $\phi(\cdot)$. Second, the upper equation represents the output of a multilayer perceptron described as follows:

1. The network has n input nodes and a single hidden layer consisting of M neurons; the inputs are denoted by $x_1, \dots, x_N$.
2. i th hidden neuron has synaptic weights $\omega_{i1}, \dots, \omega_{iN}$ and threshold θ_i .
3. The network output y_j is a linear combination of the outputs of the hidden neurons, with $\alpha_i, \dots, \alpha_M$ defining the coefficients of this combination.

The theorem actually states that a single hidden layer is sufficient for a multilayer perceptron to compute a uniform ϵ approximation to a given training set represented by the set of inputs $x_1, \dots, x_N$ and desired (target) output $f(x_1, \dots, x_N)$. However, the theorem does not say that a single layer is *optimum* in the sense of learning time or ease of implementation.

Recall that training of multilayer perceptrons is usually performed using a certain clone of the BP algorithm (2.4.2). In this forward-pass/backward-pass gradient-descending algorithm, the adjusting of synaptic weights is defined by the extended δ -rule, given by equation

$$\Delta\omega_{ji}(N) = \eta \cdot \delta_j(N) \cdot y_i(N), \quad (2.116)$$

where $\Delta\omega_{ji}(N)$ corresponds to the *weight correction*, η is the *learning-rate parameter*, $\delta_j(N)$ denotes the *local gradient* and $y_i(N)$ – the *input signal of neuron j* ; while the *cost function* E is defined as the instantaneous sum of squared errors e_j^2

$$E(n) = \frac{1}{2} \sum_j e_j^2(N) = \frac{1}{2} \sum_j [d_j(N) - y_j(N)]^2, \quad (2.117)$$

where $y_j(N)$ is the output of j th neuron, and $d_j(N)$ is the desired (target) response for that neuron. The slow BP convergence rate (2.116–2.117) can be accelerated using the faster LM algorithm (see subsection 2.4.2 above), while its robustness can be achieved using an appropriate fuzzy controller (see subsection (2.6.3) below).

Summary of Supervised Learning Methods

Gradient Descent Method

Given the $(D + 1)$ D weights vector $\mathbf{w}(n) = [w_0(n), \dots, w_D(n)]^T$ (with $w_0 = \text{bias}$), and the correspondent MSE-gradient (including partials of MSE w.r.t. weights)

$$\nabla \mathbf{e} = \left[\frac{\partial e}{\partial w_0}, \dots, \frac{\partial e}{\partial w_D} \right]^T,$$

and the learning rate (step size) η , we have the vector learning equation

$$\mathbf{w}(n + 1) = \mathbf{w}(n) - \eta \nabla \mathbf{e}(n),$$

which in index form reads

$$w_i(n + 1) = w_i(n) - \eta \nabla e_i(n).$$

LMS Algorithm

$$\mathbf{w}(n + 1) = \mathbf{w}(n) + \eta \varepsilon(n) x(n),$$

where x is an input (measurement) vector, and ε is a zero-mean Gaussian noise vector uncorrelated with input, or

$$w_i(n + 1) = w_i(n) + \eta \varepsilon(n) x^i(n).$$

Newton's Method

$$\mathbf{w}(n + 1) = \mathbf{w}(n) - \eta \mathbf{R}^{-1} \mathbf{e}(n),$$

where $\mathbf{R}$ is input (auto)correlation matrix, or

$$\mathbf{w}(n + 1) = \mathbf{w}(n) + \eta \mathbf{R}^{-1} \varepsilon(n) x(n),$$

Conjugate Gradient Method

$$\begin{aligned} \mathbf{w}(n + 1) &= \mathbf{w}(n) + \eta \mathbf{p}(n), \\ \mathbf{p}(n) &= -\nabla \mathbf{e}(n) + \beta(n) \mathbf{p}(n - 1), \\ \beta(n) &= \frac{\nabla \mathbf{e}(n)^T \nabla \mathbf{e}(n)}{\nabla \mathbf{e}(n - 1)^T \nabla \mathbf{e}(n - 1)}. \end{aligned}$$

Levenberg–Marquardt Algorithm

Putting

$$\nabla \mathbf{e} = \mathbf{J}^T \mathbf{e},$$

where $\mathbf{J}$ is the Jacobian matrix, which contains first derivatives of the network errors with respect to the weights and biases, and $\mathbf{e}$ is a vector of network errors, LM algorithm reads

$$\mathbf{w}(n+1) = \mathbf{w}(n) - [\mathbf{J}^T \mathbf{J} + \mu \mathbf{I}]^{-1} \mathbf{J}^T \mathbf{e}. \quad (2.118)$$

Other Standard ANNs*Generalized Feedforward Nets*

The *generalized feedforward network* (GFN, see Figure 2.39) is a generalization of MLP, such that connections can jump over one or more layers, which in practice, often solves the problem much more efficiently than standard MLPs. A classic example of this is the two-spiral problem, for which standard MLP requires hundreds of times more training epochs than the generalized feedforward network containing the same number of processing elements. Both MLPs and GFNs are usually trained using a variety of backpropagation techniques and their enhancements like the nonlinear LM algorithm (2.118). During training in the spatial processing, the weights of the GFN converge iteratively to the analytical solution of the 2D Laplace equation.

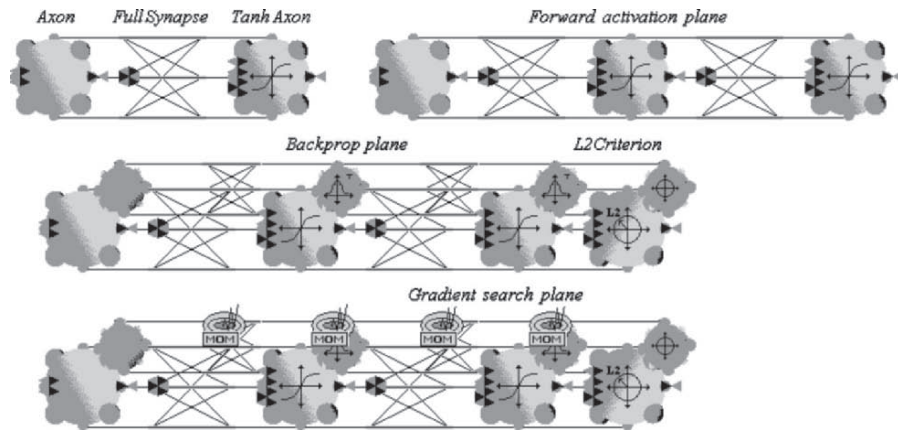


Fig. 2.39. Generalized feedforward network (GFN), arranged using *NeuroSolutions*TM.

Modular Feedforward Nets

The *modular feedforward networks* are a special class of MLP. These networks process their input using several parallel MLPs, and then recombine the results. This tends to create some structure within the topology, which will foster specialization of function in each submodule (see Figure 2.40). In contrast to the MLP, modular networks do not have full inter-connectivity between their layers. Therefore, a smaller number of weights are required for the same size network (i.e., the same number of PEs). This tends to speed up training times and reduce the number of required training exemplars. There are many ways to segment a MLP into modules. It is unclear how to best design the modular topology based on the data. There are no guarantees that each module is specializing its training on a unique portion of the data.

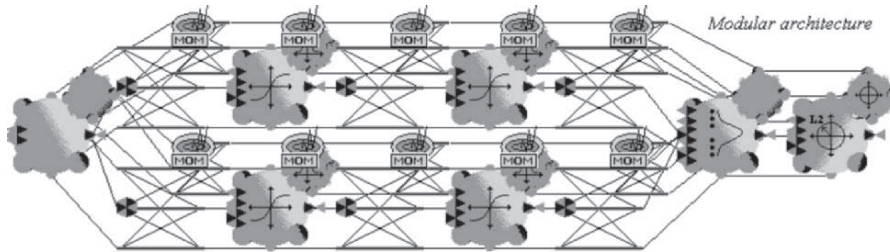


Fig. 2.40. Modular feedforward network, arranged using *NeuroSolutionsTM*.

Jordan and Elman Nets

Jordan and Elman networks (see [Elm90]) extend the multilayer perceptron with context units, which are processing elements (PEs) that remember past activity. Context units provide the network with the ability to extract temporal information from the data. In the Elman network, the activity of the first hidden PEs are copied to the context units, while the Jordan network copies the output of the network (see Figure 2.41). Networks which feed the input and the last hidden layer to the context units are also available.

Kohonen Self-Organizing Map

Kohonen self-organizing map (SOM, see Figure 2.42) is widely used for image pre-processing as well as a pre-processing unit for various hybrid architectures. SOM is a winner-take-all neural architecture that quantizes the input space, using a distance metric, into a discrete feature output space, where neighboring regions in the input space are neighbors in the discrete output space. SOM is usually applied to neighborhood clustering of random points along a circle using a variety of distance metrics: Euclidean, L^1 , L^2 , and L^n ,

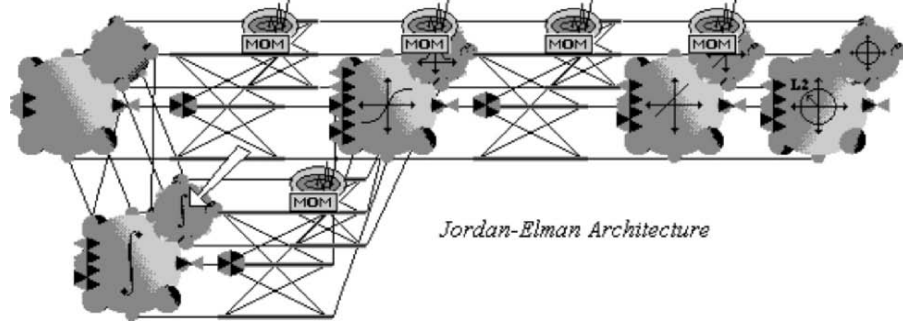


Fig. 2.41. Jordan and Elman network, arranged using *NeuroSolutions*TM.

Machalanobis, etc. The basic SOM architecture consists of a layer of Kohonen synapses of three basic forms: line, diamond and box, followed by a layer of winner-take-all axons. It usually uses added Gaussian and uniform noise, with control of both the mean and variance. Also, SOM usually requires choosing the proper initial neighborhood width as well as annealing of the neighborhood width during training to ensure that the map globally represents the input space.

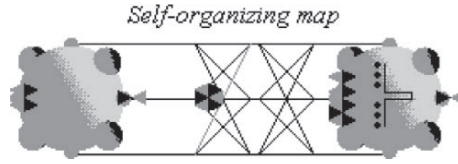


Fig. 2.42. Kohonen self-organizing map (SOM) network, arranged using *NeuroSolutions*TM.

The Kohonen SOM algorithm is defined as follows: Every stimulus $\mathbf{v}$ of an Euclidian input space V is mapped to the neuron with the position $\mathbf{s}$ in the neural layer R with the highest neural activity, the ‘center of excitation’ or ‘winner’, given by the condition

$$|\mathbf{w}_{\mathbf{s}} - \mathbf{v}| = \min_{\mathbf{r} \in R} |\mathbf{w}_{\mathbf{r}} - \mathbf{v}|,$$

where $|\cdot|$ denotes the Euclidian distance in input space. In the Kohonen model the learning rule for each synaptic weight vector $\mathbf{w}_{\mathbf{r}}$ is given by

$$\mathbf{w}_{\mathbf{r}}^{\text{new}} = \mathbf{w}_{\mathbf{r}}^{\text{old}} + \eta \cdot g_{\mathbf{rs}} \cdot (\mathbf{v} - \mathbf{w}_{\mathbf{r}}^{\text{old}}), \quad (2.119)$$

with $g_{\mathbf{rs}}$ as a gaussian function of Euclidian distance $|\mathbf{r} - \mathbf{s}|$ in the neural layer. Topology preservation is enforced by the common update of all weight

vectors whose neuron $\mathbf{r}$ is adjacent to the center of excitation $\mathbf{s}$. The function $g_{\mathbf{rs}}$ describes the topology in the neural layer. The parameter η determines the speed of learning and can be adjusted during the learning process.

Radial Basis Function Nets

The *radial basis function network* (RBF, see Figure 2.43) provides a powerful alternative to MLP for function approximation or classification. It differs from MLP in that the overall input-output map is constructed from local contributions of a layer of Gaussian axons. It trains faster and requires fewer training samples than MLP, using the hybrid supervised/unsupervised method. The unsupervised part of an RBF network consists of a competitive synapse followed by a layer of Gaussian axons. The means of the Gaussian axons are found through competitive clustering and are, in fact, the weights of the Conscience synapse. Once the means converge the variances are calculated based on the separation of the means and are associated with the Gaussian layer. Having trained the unsupervised part, we now add the supervised part, which consists of a single-layer MLP with a soft-max output.

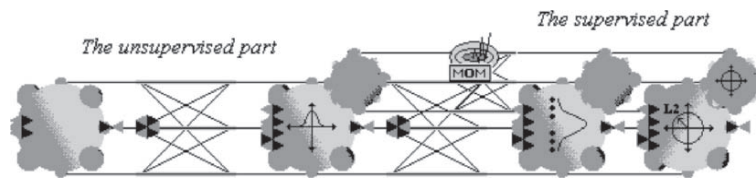


Fig. 2.43. Radial basis function network, arranged using *NeuroSolutions*TM.

Principal Component Analysis Nets

The *principal component analysis networks* (PCAs, see Figure 2.44) combine unsupervised and supervised learning in the same topology. Principal component analysis is an unsupervised linear procedure that finds a set of uncorrelated features, principal components, from the input. A MLP is supervised to perform the nonlinear classification from these components. More sophisticated are the *independent component analysis networks* (ICAs).

Co-active Neuro-Fuzzy Inference Systems

The *co-active neuro-fuzzy inference system* (CANFIS, see Figure 2.45), which integrates adaptable fuzzy inputs with a modular neural network to rapidly and accurately approximate complex functions. Fuzzy-logic inference systems (see next section) are also valuable as they combine the explanatory nature of rules (membership functions) with the power of ‘black box’ neural networks.

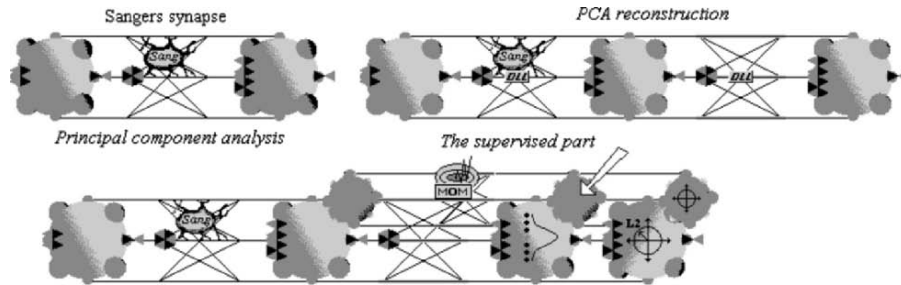


Fig. 2.44. Principal component analysis (PCA) network, arranged using *NeuroSolutionsTM*.

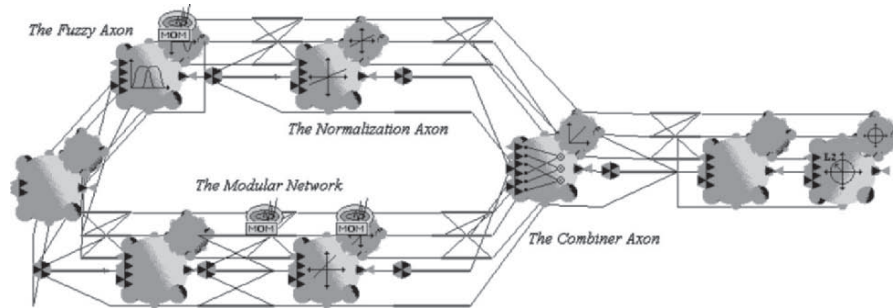


Fig. 2.45. Co-active neuro-fuzzy inference system (CANFIS) network, arranged using *NeuroSolutionsTM*.

Support Vector Machines

The *support vector machine* (SVM, see Figure 2.46 as well as next subsection), implementing the statistical learning theory, is used as the most powerful classification and decision-making system. SVMs are a radically different type of classifier that has attracted a great deal of attention lately due to the novelty of the concepts that they bring to pattern recognition, their strong mathematical foundation, and their excellent results in practical problems. SVM represents the coupling of the following two concepts: the idea that transforming the data into a high-dimensional space makes linear discriminant functions practical, and the idea of large margin classifiers to train the MLP or RBF. It is another type of a kernel classifier: it places Gaussian kernels over the data and linearly weights their outputs to create the system output. To implement the SVM-methodology, we can use the Adatron-kernel algorithm, a sophisticated nonlinear generalization of the RBF networks, which maps inputs to a high-dimensional feature space, and then optimally separates data into their respective classes by isolating those inputs, which fall close to the data boundaries. Therefore, the Adatron-kernel is especially effective in separating sets of data, which share complex boundaries, as well as for the training

for nonlinearly separable patterns. The support vectors allow the network to rapidly converge on the data boundaries and consequently classify the inputs.

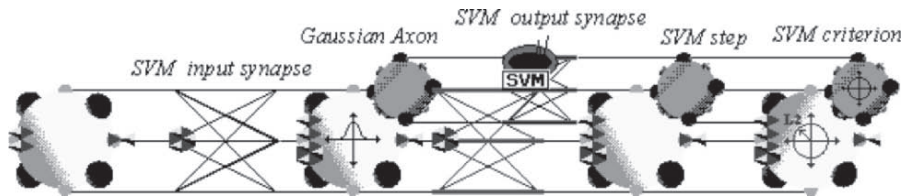


Fig. 2.46. Adatron-kernel based support vector machine (SVM) network, arranged using *NeuroSolutions*TM.

The main advantage of SVMs over MLPs is that the learning task is a *convex optimization problem* which can be reliably solved even when the example data require the fitting of a very complicated function [Vap95, Vap98]. A common argument in computational learning theory suggests that it is dangerous to utilize the full flexibility of the SVM to learn the training data perfectly when these contain an amount of noise. By fitting more and more noisy data, the machine may implement a rapidly oscillating function rather than the smooth mapping which characterizes most practical learning tasks. Its prediction ability could be no better than random guessing in that case. Hence, modifications of SVM training [CT00] that allow for training errors were suggested to be necessary for realistic noisy scenarios. This has the drawback of introducing extra model parameters and spoils much of the original elegance of SVMs.

Mathematics of SVMs is based on real *Hilbert space* methods.

Genetic ANN-Optimization

Genetic optimization, added to ensure and speed-up the convergence of all other ANN-components, is a powerful tool for enhancing the efficiency and effectiveness of a neural network. Genetic optimization can fine-tune network parameters so that network performance is greatly enhanced. Genetic control applies a *genetic algorithm* (GA, see next section), a part of broader *evolutionary computation*, see MIT journal with the same name) to any network parameters that are specified. Also through the *genetic control*, GA parameters such as mutation probability, crossover type and probability, and selection type can be modified.

Time-Lagged Recurrent Nets

The *time-lagged recurrent networks* (TLRNs, see Figure 2.47) are MLPs extended with short term memory structures [Wer90]. Most real-world data

contains information in its time structure, i.e., how the data changes with time. Yet, most neural networks are purely static classifiers. TLRNs are the state of the art in nonlinear time series prediction, system identification and temporal pattern classification. Time-lagged recurrent nets usually use memory Axons, consisting of IIR filters with local adaptable feedback that act as a variable memory depth. The time-delay neural network (TDNN) can be considered a special case of these networks, examples of which include the Gamma and Laguerre structures. The Laguerre axon uses locally recurrent all-pass IIR filters to store the recent past. They have a single adaptable parameter that controls the memory depth. Notice that in addition to providing memory for the input, we have also used a Laguerre axon after the hidden Tanh axon. This further increases the overall memory depth by providing memory for that layer's recent activations.

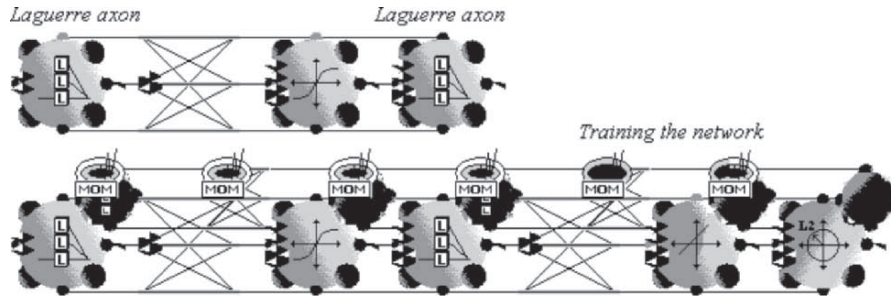


Fig. 2.47. Time-lagged recurrent network (TLRN), arranged using *NeuroSolutionsTM*.

Fully Recurrent ANNs

The *fully recurrent networks* feed back the hidden layer to itself. Partially recurrent networks start with a fully recurrent net and add a feedforward connection that bypasses the recurrency, effectively treating the recurrent part as a state memory. These recurrent networks can have an infinite memory depth and thus find relationships through time as well as through the instantaneous input space. Most real-world data contains information in its time structure. Recurrent networks are the state of the art in nonlinear time series prediction, system identification, and temporal pattern classification. In case of large number of neurons, here the firing states of the neurons or their membrane potentials are the microscopic stochastic dynamical variables, and one is mostly interested in quantities such as average state correlations and global information processing quality, which are indeed measured by macroscopic observables. In contrast to layered networks, one cannot simply write down the values of successive neuron states for models of recurrent ANNs; here

they must be solved from (mostly stochastic) coupled dynamic equations. For nonsymmetric networks, where the asymptotic (stationary) statistics are not known, dynamical techniques from non-equilibrium statistical mechanics are the only tools available for analysis. The natural set of macroscopic quantities (or order parameters) to be calculated can be defined in practice as the smallest set which will obey closed deterministic equations in the limit of an infinitely large network.

Being high-dimensional nonlinear systems with extensive feedback, the dynamics of recurrent ANNs are generally dominated by a wealth of attractors (fixed-point attractors, limit-cycles, or even more exotic types), and the practical use of recurrent ANNs (in both biology and engineering) lies in the potential for creation and manipulation of these attractors through adaptation of the network parameters (synapses and thresholds) (see [Hop82, Hop84]). Input fed into a recurrent ANN usually serves to induce a specific initial configuration (or firing pattern) of the neurons, which serves as a cue, and the output is given by the (static or dynamic) attractor which has been triggered by this cue. The most familiar types of recurrent ANN models, where the idea of creating and manipulating attractors has been worked out and applied explicitly, are the so-called attractor, associative memory ANNs, designed to store and retrieve information in the form of neuronal firing patterns and/or sequences of neuronal firing patterns. Each pattern to be stored is represented as a microscopic state vector. One then constructs synapses and thresholds such that the dominant attractors of the network are precisely the pattern vectors (in the case of static recall), or where, alternatively, they are trajectories in which the patterns are successively generated microscopic system states. From an initial configuration (the cue, or input pattern to be recognized) the system is allowed to evolve in time autonomously, and the final state (or trajectory) reached can be interpreted as the pattern (or pattern sequence) recognized by network from the input. For such programmes to work one clearly needs recurrent ANNs with extensive ergodicity breaking: the state vector will during the course of the dynamics (at least on finite time-scales) have to be confined to a restricted region of state-space (an ergodic component), the location of which is to depend strongly on the initial conditions. Hence our interest will mainly be in systems with many attractors. This, in turn, has implications at a theoretical/mathematical level: solving models of recurrent ANNs with extensively many attractors requires advanced tools from disordered systems theory, such as replica theory (statics) and generating functional analysis (dynamics).

Dynamical Games and Recurrent ANNs

IPD Games

Recall that *Prisoners' Dilemma* (PD) *game*²⁴ gets its name from the scenario where two people are arrested for some crime and are questioned separately. Each is given the opportunity to either cooperate with his/her accomplice and not give any information to the police, or defect against his/her partner by ratting to the police in exchange for some kind of immunity. The PD game is constructed to mimic this situation by providing payoffs to the players based on how they both respond. A typical payoff is to give 3 points to each if they both cooperate (police have no case against criminals and can only get them for a very light crime). If one cooperates, and the other defects, the defector gets 5 points, while his partner receives none. If both players defect, they each only receive 1 point (police get them for a moderate crime, but not as severe as if only one person had taken the blame). The total payoff to both players is greatest for mutual cooperation (6 points), while a cooperate–defect play results in 5 points, and mutual defection only hands out 2 points. So, clearly, it is best for the collective system if all players cooperate, but here is the interesting paradox. Each player individually is better off by defecting in any given situation. If your partner cooperates, you can get 3 points by cooperating back, but you can get 5 by defecting. Similarly, if your partner defects, you get nothing if you cooperate, but you can still salvage 1 point by defecting back.

The extension of the PD game to permit repeated PD game play between players is known as the *Iterated Prisoner's Dilemma* (IPD) game. In 1979, R. Axelrod hosted a tournament at the University of Michigan to see what

²⁴ Recall that *game theory*, first formulated by John von Neumann and Oskar Morgenstern first formalized the subject in 1944 [NM44] is a branch of applied mathematics that uses models to study interactions with formalized incentive structures ('games'). It has applications in a variety of fields, including economics, international relations, evolutionary biology, political science, and military strategy. Game theorists study the predicted and actual behavior of individuals in games, as well as optimal strategies. Seemingly different types of interactions can exhibit similar incentive structures, thus all exemplifying one particular game. The Prisoner's Dilemma game, as popularized by mathematician A. Tucker, furnishes an example of the application of game theory to real life; it has many implications for the nature of human co-operation. Another important concept in game theory is the *Nash equilibrium*, formulated by *Nobel Laureate John Nash* in his dissertation on Non-Cooperative Games in 1950. Nash showed that the various solutions for games that had been given earlier all yield Nash equilibria. For example, the Prisoner's dilemma has one Nash equilibrium: when both players defect. However, 'both defect' is inferior to 'both cooperate', in the sense that the total jail time served by the two prisoners is greater if both defect. The strategy 'both cooperate' is unstable, as a player could do better by defecting while their opponent still cooperates. Thus, 'both cooperate' is not an equilibrium.

kinds of strategies would perform best over the long haul in the IPD game [Axe76]. Various game theorists were solicited for IPD strategies in the form of computer algorithms, and the submitted strategies were then pitted against each other in repeated round-robin PD game play. The strategy that received the highest total payoff from this repeated PD game play was declared the winner of the tournament.

Recurrent ANN Model for a Two-Player IPD Game

A two-player IPD game was modeled using a *recurrent ANN* in [TI99]. The authors created two *internal models* of game players. The first one was ‘pure reductionist Bob’, who made the opponent’s model by a recurrent ANN. He thought that the opponent might behave with simple algorithms like finite automata, which could be expressed by the recurrent nets. The second one was ‘clever Alice’, who assumed that the opponent behaved like ‘pure reductionist Bob’. She knew that the opponent made her model by recurrent nets and she decided the next action based on that model of herself. In other words, she builded the internal model of herself and treated that model as her image in the opponent.

Building the model of a player’s behavior may involve many methods such as finite automata, Markov chain, etc. In [TI99] the authors used *dynamical recognizers*, simple and powerful tools for studying dynamical behavior from the view points of cognitive studies. Dynamical recognizers were first discussed by Pollack [Pol91], who showed that some automata could be learned very well by dynamical recognizers. When it could not learn automata, fractal-like patterns were generated in context spaces.

The so-called *cascaded recurrent ANN* consists of a function and a context net [Pol91]. It is quite similar to a two-layer linear perceptron, though the recurrent outputs are feeded back not to the input nodes but to the weights of the function network. The recurrent outputs memorize the opponent’s current status, and the context network converts the recurrent outputs to the weights of the function network which predicts the next action. The space constructed by the outputs from the function network (including both recurrent and network outputs) was called the *context space*. The output was taken from a node of the function network. In [TI99], only one input and one output node were necessary since the IPD game had only two actions, cooperation and defection. The authors defined cooperation as 0 and defection as 1 in the network. The output was rounded off to 0 (cooperation) and 1 (defection). The network is described by the following equations,

$$z_i(n) = g(w_i y(n) + w_i^0), \quad w_i = u_{ij} z_j(n-1) + u_i^b, \quad w_i^0 = u_{ij}^0 z_j(n-1) + u_i^{0b},$$

where symbols have the following meanings: $g(x) = (e^{-x} + 1)^{-1}$ is the sigmoid function, $y(n)$ is the input, $z_0(n)$ is the output, $z_i(n)$, ($i = 1 \cdots N$) are the recurrent outputs, w_i is the weight of function network, w_i^0 is the bias of

function network, u_{ij}, u_{ij}^0 is the weight of context network and u_i^b, u_i^{0b} is the bias of context network.

This recurrent ANN was trained by a form of the back-propagation method. In the game, the player knew only his or her own past actions and those of the opponent. In the case of ‘pure reductionist Bob’, the model of the opponent was built by the recurrent ANN. This means that the recurrent ANN takes the player’s last action as an input and outputs the opponent’s next action. Thus, the target for training is a series of the opponent’s action when the inputs are the player’s actions. However, since the number of training targets becomes too large as a game proceeds, the weights for learning are varied for each action so that far actions in the distant past are forgotten. Thus, the error $E(n)$ after the n th game is $E(n) = \sum_{k=1}^n \lambda^{n-k} (z_0(k) - d(k))^2$, where $d(k)$ is a target (i.e., the actual opponent’s action in the k th game), $z_0(k)$ is the predicted action by the recurrent ANN, and λ ($= 0.9$ for most simulations) is a parameter which controls the memory retention of the past actions.

To determine the player’s next action, the authors of [TI99] used the prediction of the opponent’s future action based on the recurrent ANN. First, the algorithm for pure reductionist Bob was explained. Bob chose his forward actions up to M games. Then, he could predict the opponent’s actions from his forward actions, and the expected score could be evaluated. The process was repeated for all possible strings of actions of length M and Bob chose the action with the highest score as the best action.

Clever Alice considered the opponent was a pure reductionist Bob, so she chose her forward M actions. She predicted the opponent’s actions assuming that he behaved like pure reductionist Bob. Again the process was repeated for all strings of the length M and she chose the action string with the highest score as the best one. In other words, she predicted her image in the other person and tried to educate him to have a favorable image through her actions.

Complex-Valued ANNs

It is expected that *complex-valued ANNs*, whose parameters (weights and threshold values) are all complex numbers, will have applications in all the fields dealing with complex numbers (e.g., telecommunications, quantum physics). A complex-valued, feedforward, multi-layered, back-propagation neural network model was proposed independently by [NF91, Nit97, Nit00, Nit04], [GK92] and [BP92], and demonstrated its characteristics:

- (a) the properties greatly different from those of the real-valued back-propagation network, including 2D motion structure of weights and the orthogonality of the decision boundary of a complex-valued neuron;
- (b) the learning property superior to the real-valued back-propagation;
- (c) the inherent 2D motion learning ability (an ability to transform geometric figures); and

(d) the ability to solve the XOR problem and detection of symmetry problem with a single complex-valued neuron.

Following [NF91, Nit97, Nit00, Nit04], we consider here the complex-valued neuron. Its input signals, weights, thresholds and output signals are all complex numbers. The net input U_n to a complex-valued neuron n is defined as

$$U_n = W_{mn}X_m + V_n,$$

where W_{mn} is the (complex-valued) weight connecting the complex-valued neurons m and n , V_n is the (complex-valued) threshold value of the complex-valued neuron n , and X_m is the (complex-valued) input signal from the complex-valued neuron m . To get the (complex-valued) output signal, convert the net input U_n into its real and imaginary parts as follows: $U_n = x + iy = z$, where $i = \sqrt{-1}$. The (complex-valued) output signal is defined to be

$$\sigma(z) = \tanh(x) + i \tanh(y),$$

where $\tanh(u) = (\exp(u) - \exp(-u)) / (\exp(u) + \exp(-u))$, $u \in \mathbb{R}$. Note that $-1 < \text{Re}[\sigma]$, $\text{Im}[\sigma] < 1$. Note also that σ is not regular as a complex function, because the Cauchy-Riemann equations do not hold.

A complex-valued ANN consists of such complex-valued neurons described above. A typical network has 3 layers: $m \rightarrow n \rightarrow 1$, with $w_{ij} \in \mathbb{C}$ – the weight between the input neuron i and the hidden neuron j , $w_{0j} \in \mathbb{C}$ – the threshold of the hidden neuron j , $c_j \in \mathbb{C}$ – the weight between the hidden neuron j and the output neuron ($1 \leq i \leq m; 1 \leq j \leq n$), and $c_0 \in \mathbb{C}$ – the threshold of the output neuron. Let $y_j(z), h(z)$ denote the output values of the hidden neuron j , and the output neuron for the input pattern $z = [z_1, \dots, z_m]^t \in \mathbb{C}^m$, respectively. Let also $\nu_j(z)$ and $\mu(z)$ denote the net inputs to the hidden neuron j and the output neuron for the input pattern $z \in \mathbb{C}^m$, respectively. That is,

$$\begin{aligned} \nu_j(z) &= w_{ij}z_i + w_{0j}, & \mu(z) &= c_j y_j(z) + c_0, \\ y_j(z) &= \sigma(\nu_j(z)), & h(z) &= \sigma(\mu(z)). \end{aligned}$$

The set of all $m \rightarrow n \rightarrow 1$ complex-valued ANNs described above is usually denoted by $N_{m,n}$. The Complex-BP learning rule [NF91, Nit97, Nit00, Nit04] has been obtained by using a steepest-descent method for such (multilayered) complex-valued ANNs.

2.4.3 Common Continuous ANNs

Virtually all computer-implemented ANNs (mainly listed above) are discrete dynamical systems, mainly using supervised training (except Kohonen SOM) in one of gradient-descent searching forms. They are good as problem-solving tools, but they fail as models of animal nervous system. The other category of ANNs are continuous neural systems that can be considered as models of animal nervous system. However, *as models of the human brain, all current ANNs are simply trivial.*

Neurons as Functions

According to B. Kosko, neurons behave as functions [Kos92]; they transduce an unbounded input *activation* $x(t)$ into output *signal* $S(x(t))$. Usually a sigmoidal (S-shaped, bounded, monotone-nondecreasing: $S' \geq 0$) function describes the transduction, as well as the input-output behavior of many operational amplifiers. For example, the *logistic signal* (or, the *maximum-entropy*) function

$$S(x) = \frac{1}{1 + e^{-cx}}$$

is sigmoidal and strictly increases for positive scaling constant $c > 0$. Strict monotonicity implies that the *activation derivative* of S is positive:

$$S' = \frac{dS}{dx} = cS(1 - S) > 0.$$

An infinitely steep logistic signal function gives rise to a threshold signal function

$$S(x^{n+1}) = \begin{cases} 1, & \text{if } x^{n+1} > T, \\ S(x^n), & \text{if } x^{n+1} = T, \\ 0, & \text{if } x^{n+1} < T, \end{cases}$$

for an arbitrary real-valued threshold T . The index n indicates the discrete time step.

In practice signal values are usually binary or bipolar. *Binary signals*, like logistic, take values in the unit interval $[0, 1]$. *Bipolar signals* are signed; they take values in the bipolar interval $[-1, 1]$. Binary and bipolar signals transform into each other by simple scaling and translation. For example, the bipolar logistic signal function takes the form

$$S(x) = \frac{2}{1 + e^{-cx}} - 1.$$

Neurons with bipolar threshold signal functions are called *McCulloch-Pitts neurons*.

A naturally occurring bipolar signal function is the *hyperbolic-tangent* signal function

$$S(x) = \tanh(cx) = \frac{e^{cx} - e^{-cx}}{e^{cx} + e^{-cx}},$$

with activation derivative

$$S' = c(1 - S^2) > 0.$$

The *threshold linear* function is a binary signal function often used to approximate neuronal firing behavior:

$$S(x) = \begin{cases} 1, & \text{if } cx \geq 1, \\ 0, & \text{if } cx < 0, \\ cx, & \text{else,} \end{cases}$$

which we can rewrite as

$$S(x) = \min(1, \max(0, cx)).$$

Between its upper and lower bounds the threshold linear signal function is trivially monotone increasing, since $S' = c > 0$.

Gaussian, or bell-shaped, signal function of the form $S(x) = e^{-cx^2}$, for $c > 0$, represents an important exception to signal monotonicity. Its activation derivative $S' = -2cxe^{-cx^2}$ has the sign opposite the sign of the activation x .

Generalized Gaussian signal functions define potential or radial basis functions $S_i(x^i)$ given by

$$S_i(x) = \exp\left[-\frac{1}{2\sigma_i^2} \sum_{j=1}^n (x_j - \mu_j^i)^2\right],$$

for input activation vector $x = (x^i) \in \mathbb{R}^n$, variance σ_i^2 , and mean vector $\mu_i = (\mu_j^i)$. Each radial basis function S_i defines a spherical *receptive field* in $\mathbb{R}^n$. The i th neuron emits unity, or near-unity, signals for sample activation vectors x that fall in its receptive field. The mean vector μ centers the receptive field in $\mathbb{R}^n$. The variance σ_i^2 localizes it. The radius of the Gaussian spherical receptive field shrinks as the variance σ_i^2 decreases. The receptive field approaches $\mathbb{R}^n$ as σ_i^2 approaches ∞ .

The *signal velocity* $\dot{S} \equiv dS/dt$ is the *signal time derivative*, related to the activation derivative by

$$\dot{S} = S' \dot{x},$$

so it depends explicitly on *activation velocity*. This is used in unsupervised learning laws that adapt with *locally available information*.

The signal $S(x)$ induced by the activation x represents the neuron's firing frequency of action potentials, or pulses, in a sampling interval. The firing frequency equals the average number of pulses emitted in a sampling interval.

Short-term memory is modeled by *activation dynamics*, and *long-term memory* is modeled by *learning dynamics*. The overall neural network behaves as an *adaptive filter* (see [Hay91]).

In the simplest and most common case, neurons are not topologically ordered. They are related only by the synaptic connections between them. Kohonen calls this *lack of topological structure* in a *field of neurons* the *zeroth-order topology*. This suggests that ANN-models are *abstractions*, not *descriptions* of the brain neural networks, in which order does matter.

Basic Activation and Learning Dynamics

One of the oldest continuous training methods, based on Hebb's biological synaptic learning [Heb49], is *Oja-Hebb learning rule* [Oja82], which calculates the weight update according to the ODE

$$\dot{\omega}_i(t) = O(t) [I_i(t) - O(t) \omega_i(t)],$$

where $O(t)$ is the output of a simple, linear processing element; $I_i(t)$ are the inputs; and $\omega_i(t)$ are the synaptic weights.

Related to the Oja–Hebb rule is a special matrix of synaptic weights called *Karhunen–Loeve covariance matrix* $\mathbf{W}$ (KL), with entries

$$W_{ij} = \frac{1}{N} \omega_i^\mu \omega_j^\mu, \quad (\text{summing over } \mu)$$

where N is the number of vectors, and ω_i^μ is the i th component of the μ th vector. The KL matrix extracts the principal components, or directions of maximum information (correlation) from a dataset.

In general, continuous ANNs are *temporal dynamical systems*. They have two coupled dynamics: activation and learning. First, a general system of coupled ODEs for the output of the i th *processing element* (PE) x^i , called the *activation dynamics*, can be written as

$$\dot{x}^i = g_i(x^i, \text{net}_i), \quad (2.120)$$

with the *net input* to the i th PE x^i given by $\text{net}_i = \omega_{ij} x^j$.

For example,

$$\dot{x}^i = -x^i + f_i(\text{net}_i),$$

where f_i is called *output*, or *activation*, *function*. We apply some input values to the PE so that $\text{net}_i > 0$. If the inputs remain for a sufficiently long time, the output value will reach an equilibrium value, when $\dot{x}^i = 0$, given by $x^i = f_i(\text{net}_i)$. Once the unit has a nonzero output value, removal of the inputs will cause the output to return to zero. If $\text{net}_i = 0$, then $\dot{x}^i = -x^i$, which means that $x \rightarrow 0$.

Second, a general system of coupled ODEs for the *update* of the synaptic weights ω_{ij} , i.e., *learning dynamics*, can be written as a generalization of the Oja–Hebb rule, i.e.,

$$\dot{\omega}_{ij} = G_i(\omega_{ij}, x^i, x^i),$$

where G_i represents the *learning law*; the learning process consists of finding weights that encode the knowledge that we want the system to learn. For most realistic systems, it is not easy to determine a closed-form solution for this system of equations, so the approximative solutions are usually enough.

Standard Models of Continuous Nets

Hopfield Continuous Net

One of the first physically-based ANNs was developed by J. Hopfield. He first made a discrete, Ising–spin based network in [Hop82], and later generalized it to the continuous, graded-response network in [Hop84], which we briefly

describe here. Later we will give full description of Hopfield models. Let $\text{net}_i = u_i$ – the net input to the i th PE, biologically representing the summed action potentials at the axon hillock of a neuron. The PE *output function* is

$$v_i = g_i(\lambda u_i) = \frac{1}{2}(1 + \tanh(\lambda u_i)),$$

where λ is a constant called the *gain parameter*. The network is described as a transient RC circuit

$$C_i \dot{u}_i = T_{ij} v_j - \frac{u_i}{R_i} + I_i, \quad (2.121)$$

where I_i, R_i and C_i are inputs (currents), resistances and capacitances, and T_{ij} are synaptic weights.

The Hamiltonian energy function corresponding to (2.121) is given as

$$H = -\frac{1}{2} T_{ij} v_i v_j + \frac{1}{\lambda} \frac{1}{R_i} \int_0^{v_i} g_i^{-1}(v) dv - I_i v_i, \quad (j \neq i) \quad (2.122)$$

which is a generalization of a discrete, *Ising-spin Hopfield network* with energy function

$$E = -\frac{1}{2} \omega_{ij} x^i x^j, \quad (j \neq i).$$

where $g_i^{-1}(v) = u$ is the inverse of the function $v = g(u)$. To show that (2.122) is an appropriate *Lyapunov function* for the system, we shall take its time derivative assuming T_{ij} are symmetric:

$$\dot{H} = -\dot{v}_i (T_{ij} v_j - \frac{u_i}{R_i} + I_i) = -C_i \dot{v}_i \dot{u}_i = -C_i \dot{v}_i^2 \frac{\partial g_i^{-1}(v_i)}{\partial v_i}. \quad (2.123)$$

All the factors in the summation (2.123) are positive, so $\dot{H}$ must decrease as the system evolves, until it eventually reaches the stable configuration, where $\dot{H} = \dot{v}_i = 0$. For further development on Hopfield nets, see section (3.1.2) below.

Hecht-Nielsen Counterpropagation Net

Hecht-Nielsen counterpropagation network (CPN) is a full-connectivity, graded-response generalization of the standard BP algorithm (see [Hec87, Hec90]). The outputs of the PEs in CPN are governed by the set of ODEs

$$\dot{x}^i = -A x_i + (B - x^i) I_i - x^i \sum_{j \neq i} I_j,$$

where $0 < x^i(0) < B$, and $A, B > 0$. Each PE receives a net excitation (on-center) of $(B - x^i) I_i$ from its corresponding input value, I . The addition of inhibitory connections (off-surround), $-x^i I_j$, from other units is responsible for preventing the activity of the processing element from rising in proportion

to the absolute pattern intensity, I_i . Once an input pattern is applied, the PEs quickly reach an equilibrium state ($\dot{x}^i = 0$) with

$$x^i = \Theta_i \frac{BI_i}{A + I_i},$$

with the normalized *reflectance pattern* $\Theta_i = I_i (\sum_i I_i)^{-1}$, such that $\sum_i \Theta_i = 1$.

Competitive Net

Activation dynamics is governed by the ODEs

$$\dot{x}^i = -Ax_i + (B - x^i)[f(x^i) + \text{net}_i] - x^i \left[\sum_{j \neq i} f(x_j) + \sum_{j \neq i} \text{net}_j \right],$$

where $A, B > 0$ and $f(x^i)$ is an output function.

Kohonen's Continuous SOM and Adaptive Robotics Control

Kohonen continuous self organizing map (SOM) is actually the original Kohonen model of the biological neural process (see [Koh88]). SOM activation dynamics is governed by

$$\dot{x}^i = -r_i(x^i) + \text{net}_i + z_{ij}x_j, \quad (2.124)$$

where the function $r_i(x^i)$ is a general form of a loss term, while the final term models the lateral interactions between units (the sum extends over all units in the system). If z_{ij} takes the form of the Mexican-hat function, then the network will exhibit a bubble of activity around the unit with the largest value of net input.

SOM learning dynamics is governed by

$$\dot{\omega}_{ij} = \alpha(t)(I_i - \omega_{ij})U(x^i),$$

where $\alpha(t)$ is the learning momentum, while the function $U(x^i) = 0$ unless $x^i > 0$ in which case $U(x^i) = 1$, ensuring that only those units with positive activity participate in the learning process.

Kohonen's continuous SOM (2.124–2.4.3) is widely used in adaptive robotics control. Having an n -segment robot arm with n chained $SO(2)$ -joints, for a particular initial position x and desired velocity $\dot{x}_{desir}^j$ of the end-effector, the required torques T_i in the joints can be found as

$$T_i = a_{ij} \dot{x}_{desir}^j,$$

where the inertia matrix $a_{ij} = a_{ij}(x)$ is learned using SOM.

Adaptive Resonance Theory

Principles derived from an analysis of experimental literatures in vision, speech, cortical development, and reinforcement learning, including attentional blocking and cognitive-emotional interactions, led to the introduction of S. Grossberg's *adaptive resonance theory* (ART) as a theory of human *cognitive information processing* (see [CG03]). The theory has evolved as a series of real-time neural network models that perform unsupervised and supervised learning, pattern recognition, and prediction. Models of unsupervised learning include ART1, for binary input patterns, and fuzzy-ART and ART2, for analog input patterns [Gro82, CG03]. ARTMAP models combine two unsupervised modules to carry out supervised learning. Many variations of the basic supervised and unsupervised networks have since been adapted for technological applications and biological analyzes.

A central feature of all ART systems is a *pattern matching process* that compares an external input with the internal memory of an active code. ART matching leads either to a resonant state, which persists long enough to permit learning, or to a parallel memory search. If the search ends at an established code, the memory representation may either remain the same or incorporate new information from matched portions of the current input. If the search ends at a new code, the memory representation learns the current input. This match-based learning process is the foundation of ART *code stability*. Match-based learning allows memories to change only when input from the external world is close enough to internal expectations, or when something completely new occurs. This feature makes ART systems well suited to problems that require on-line learning of large and evolving databases (see [CG03]).

Many ART applications use fast learning, whereby adaptive weights converge to equilibrium in response to each input pattern. Fast learning enables a system to adapt quickly to inputs that occur rarely but that may require immediate accurate recall. Remembering details of an exciting movie is a typical example of learning on one trial. Fast learning creates memories that depend upon the order of input presentation. Many ART applications exploit this feature to improve accuracy by voting across several trained networks, with voters providing a measure of confidence in each prediction.

Match-based learning is complementary to *error-based learning*, which responds to a mismatch by changing memories so as to reduce the difference between a target output and an actual output, rather than by searching for a better match. Error-based learning is naturally suited to problems such as adaptive control and the learning of *sensory-motor maps*, which require ongoing adaptation to present statistics. Neural networks that employ error-based learning include backpropagation and other multilayer perceptrons (MLPs).

Activation dynamics of ART2 is governed by the ODEs [Gro82, CG03]

$$\epsilon \dot{x}_i = -Ax_i + (1 - Bx_i)I_i^+ - (C + Dx_i)I_i^-,$$

where ϵ is the ‘small parameter’, I_i^+ and I_i^- are excitatory and inhibitory inputs to the i th unit, respectively, and $A, B, C, D > 0$ are parameters.

General *Cohen–Grossberg activation equations* have the form:

$$\dot{v}_j = -a_j(v_j)[b_j(v_j) - f_k(v_k)m_{jk}], \quad (j = 1, \dots, N), \quad (2.125)$$

and the *Cohen–Grossberg theorem* ensures the global stability of the system (2.125). If

$$a_j = 1/C_j, b_j = v_j/R_j - I_j, f_j(v_j) = u_j,$$

and constant $m_{ij} = m_{ji} = T_{ji}$, the system (2.125) reduces to the Hopfield circuit model (2.121).

ART and distributed ART (dART) systems are part of a growing family of self-organizing network models that feature attentional feedback and stable code learning. Areas of technological application include industrial design and manufacturing, the control of mobile robots, face recognition, remote sensing land cover classification, target recognition, medical diagnosis, electrocardiogram analysis, signature verification, tool failure monitoring, chemical analysis, circuit design, protein/DNA analysis, 3D visual object recognition, musical analysis, and seismic, sonar, and radar recognition. ART principles have further helped explain parametric behavioral and brain data in the areas of visual perception, object recognition, auditory source identification, variable-rate speech and word recognition, and *adaptive sensory-motor control* (see [CG03]).

Spatiotemporal Networks

In *spatiotemporal networks*, activation dynamics is governed by the ODEs

$$\begin{aligned} \dot{x}^i &= A(-ax_i + b[I_i - \Gamma]^+), \\ \dot{T} &= \alpha(S - T) + \beta\dot{S}, \quad \text{with} \\ [u]^+ &= \begin{cases} u & \text{if } u > 0 \\ 0 & \text{if } u \leq 0 \end{cases}, \\ A(u) &= \begin{cases} u & \text{if } u > 0 \\ cu & \text{if } u \leq 0 \end{cases}. \end{aligned}$$

where $a, b, \alpha, \beta > 0$ are parameters, $T > 0$ is the *power-level target*, $S = \sum_i x^i$, and $A(u)$ is called the *attack function*.

Learning dynamics is given by *differential Hebbian law*

$$\begin{aligned} \dot{\omega}_{ij} &= (-c\omega_{ij} + dx_i x_j)U(\dot{x}^i)U(-\dot{x}^j), \quad \text{with} \\ U(s) &= \begin{cases} 1 & \text{if } s > 0 \\ 0 & \text{if } s \leq 0 \end{cases} \quad \text{where } c, d > 0 \text{ are constants.} \end{aligned}$$

2.4.4 Support Vector Machines

Recall that *support vector machines* (SVMs) are a set of related *supervised learning* methods used for *classification* and *regression* (see [Vap95, Vap98, SS01, CT00]). Their common factor is the use of a technique known as the ‘*kernel trick*’ to apply *linear classification* techniques to *nonlinear classification* problems.

Linear Classification Problem

Suppose we want to classify some data points into two classes. Often we are interested in classifying data as part of a machine-learning process. These data points may not necessarily be points in $\mathbb{R}^2$ but may be multidimensional $\mathbb{R}^p$ (statistics notation) or $\mathbb{R}^n$ (computer science notation) points. We are interested in whether we can separate them by a *hyperplane*. As we examine a hyperplane, this form of classification is known as linear classification. We also want to choose a hyperplane that separates the data points ‘neatly’, with maximum distance to the closest data point from both classes – this distance is called the *margin*. We desire this property since if we add another data point to the points we already have, we can more accurately classify the new point since the separation between the two classes is greater. Now, if such a hyperplane exists, the hyperplane is clearly of interest and is known as the *maximum-margin hyperplane* or the *optimal hyperplane*, as are the vectors that are closest to this hyperplane, which are called the *support vectors*.

Formalization

Consider data points of the form

$$\{(\mathbf{x}_1, c_1), (\mathbf{x}_2, c_2), \dots, (\mathbf{x}_n, c_n)\},$$

where the c_i is either 1 or -1 ; this constant denotes the class to which the point $\mathbf{x}_i$ belongs. Each $\mathbf{x}_i$ is a p D (statistics notation), or n D (computer science notation) vector of scaled $[0, 1]$ or $[-1, 1]$ values. The scaling is important to guard against variables (attributes) with larger variance that might otherwise dominate the classification. We can view this as *training data*, which denotes the correct classification which we would like the SVM to eventually distinguish, by means of the dividing hyperplane, which takes the form:

$$\mathbf{w} \cdot \mathbf{x} - b = 0.$$

As we are interested in the maximum margin, we are interested in the support vectors and the parallel hyperplanes (to the optimal hyperplane) closest to these support vectors in either class (see Figure 2.48). It can be shown that these parallel hyperplanes can be described by equations

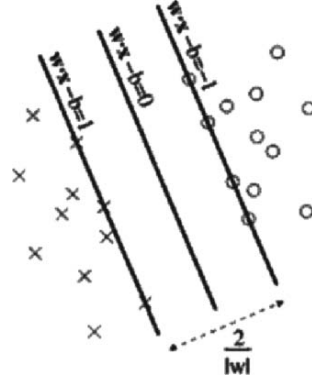


Fig. 2.48. Maximum-margin hyperplanes for a SVM trained with samples from two classes. Samples along the hyperplanes are called the support vectors.

$$\mathbf{w} \cdot \mathbf{x} - b = 1, \quad (2.126)$$

$$\mathbf{w} \cdot \mathbf{x} - b = -1. \quad (2.127)$$

We would like these hyperplanes to maximize the distance from the dividing hyperplane and to have no data points between them. By using geometry, we find the distance between the hyperplanes being $2/|\mathbf{w}|$, so we want to minimize $|\mathbf{w}|$. To exclude data points, we need to ensure that for all i either

$$\begin{aligned} \mathbf{w} \cdot \mathbf{x}_i - b &\geq 1, & \text{or} \\ \mathbf{w} \cdot \mathbf{x}_i - b &\leq -1. \end{aligned}$$

This can be rewritten as

$$c_i(\mathbf{w} \cdot \mathbf{x}_i - b) \geq 1, \quad (1 \leq i \leq n). \quad (2.128)$$

The problem now is to minimize $|\mathbf{w}|$ subject to the constraint (2.128). This is a *quadratic programming optimization* (QP) problem.

After the SVM has been trained, it can be used to classify unseen ‘test’ data. This is achieved using the following decision rule,

$$\hat{c} = \begin{cases} 1 & \text{if } \mathbf{w} \cdot \mathbf{x} + b \geq 0, \\ -1 & \text{if } \mathbf{w} \cdot \mathbf{x} + b \leq 0. \end{cases}$$

Writing the classification rule in its dual form reveals that classification is only a function of the support vectors, i.e., the training data that lie on the margin.

Vapnik–Chervonenkis SVM Theory

The use of the maximum-margin hyperplane is motivated by *Vapnik–Chervonenkis SVM theory*, which provides a probabilistic test *error bound*

that is minimized when the margin is maximized. However the utility of this theoretical analysis is sometimes questioned given the large slack associated with these bounds: the bounds often predict more than 100% error rates.

The parameters of the maximum-margin hyperplane are derived by solving the optimization. There exist several specialized algorithms for quickly solving the QP problem that arises from SVMs. The most common method for solving the QP problem is Platt's *SMO algorithm*.

Nonlinear Classification

The original optimal hyperplane algorithm proposed by Vladimir Vapnik in 1963 was a *linear classifier*. However, in 1992, B. Boser, I. Guyon and Vapnik suggested a way to create nonlinear classifiers by applying the *kernel trick* (originally proposed by Aizerman) to maximum-margin hyperplanes. The resulting algorithm is formally similar, except that every *dot product* is replaced by a nonlinear *kernel* function. This allows the algorithm to fit the maximum-margin hyperplane in the transformed feature *space*. The transformation may be nonlinear and the transformed space high dimensional; thus though the classifier is a hyperplane in the high-dimensional feature space it may be nonlinear in the original input space.

If the kernel used is a Gaussian *radial basis function*, the corresponding feature space is a *Hilbert space* of infinite dimension. Maximum margin classifiers are well *regularized*, so the infinite dimension does not spoil the results. Some common kernels include:

1. *Polynomial (homogeneous)*:

$$k(\mathbf{x}, \mathbf{x}') = (\mathbf{x} \cdot \mathbf{x}')^d;$$

2. *Polynomial (inhomogeneous)*:

$$k(\mathbf{x}, \mathbf{x}') = (\mathbf{x} \cdot \mathbf{x}' + 1)^d;$$

3. *Radial Basis Function*:

$$k(\mathbf{x}, \mathbf{x}') = \exp(-\gamma \|\mathbf{x} - \mathbf{x}'\|^2), \quad \text{for } \gamma > 0;$$

4. *Gaussian radial basis function*:

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2\sigma^2}\right); \quad \text{and}$$

5. *Sigmoid*:

$$k(\mathbf{x}, \mathbf{x}') = \tanh(\kappa \mathbf{x} \cdot \mathbf{x}' + c),$$

for some (not every) $\kappa > 0$ and $c < 0$.

Soft Margin

In 1995, *Corinna Cortes* and Vapnik suggested a modified maximum margin idea that allows for mislabeled examples. If there exists no hyperplane that can split the ‘yes’ and ‘no’ examples, the so-called *soft margin method* will choose a hyperplane that splits the examples as cleanly as possible, while still maximizing the distance to the nearest cleanly split examples. This work popularized the expression *Support Vector Machine* or *SVM*. This method introduces slack variables and the equation (2.128) now transforms to

$$c_i(\mathbf{w} \cdot \mathbf{x}_i - b) \geq 1 - \xi_i, \quad (1 \leq i \leq n), \quad (2.129)$$

and the optimization problem becomes

$$\min ||w||^2 + C \sum_i \xi_i \quad \text{such that} \quad c_i(\mathbf{w} \cdot \mathbf{x}_i - b) \geq 1 - \xi_i, \quad (1 \leq i \leq n),$$

This constraint in (2.129) along with the objective of minimizing $|w|$ can be solved using *Lagrange multipliers* or setting up a dual optimization problem to eliminate the slack variable.

SV Regression

A version of a SVM for regression was proposed in 1995 by Vapnik, S. Golowich, and A. Smola (see [Vap98, SS01]). This method is called *support vector regression* (SVR). The model produced by support vector classification (as described above) only depends on a subset of the training data, because the cost function for building the model does not care about training points that lie beyond the margin. Analogously, the model produced by SVR only depends on a subset of the training data, because the cost function for building the model ignores any training data that is close (within a threshold ε) to the model prediction.

2.4.5 Dynamical Nets for Thought Process Modeling

In this subsection, following [Gro05], we present several guiding principles for thought processes modeling, as well as an NN-type model implementing these principles is presented and studied. We consider thinking process within an AM network built-up from overlapping memory states.

The notion of ‘thinking’ comes in various flavors. We may associate thinking with logical reasoning or with a series of associative processes. The latter activity is performed effortlessly by the human brain and is at the center of the investigation we will present here. We will consider in particular associative processes which may occur also in the absence of any interaction of the brain with the outside world. It is clear that without any prior stored information, i.e., our memories, this kind of thought process would be semantically empty.

We do therefore investigate the autonomous generation of a time-series of memory-states by a cognitive system, being it natural or artificial [Gro05]. Therefore, we consider thought processes to be characterized by the spontaneous activation of one memory state by another one, leading to a history of memory states. This process should be autonomous and no outside regulative unit should be needed in order to control this dynamical process. In order to make sense, each activated memory state should be closely associated to its predecessor. A history of random memory states would not possibly classify as a true thought process.

A key question in this context is then: When can two or more memory states be considered to be closely associated? Intuitively this is not a problem: Remembering a trip to the forest with our family for a picnic we may associate this activity with a trip to the forest to cut a Christmas tree. These two memory-states are intuitively related. When we store new memory states, like the two trips to the forest in the above example, in our brain, the appropriate associative links need to be generated spontaneously. But how should our brain be capable of finding all possible relations linking this new information associatively with all previously stored memory-states? An exhaustive search would not be feasible, the time needed to perform it would be immense.

This computational problem of embedding new memory states into their relevant semantic context would not occur if no explicit associative links would be needed at all. This can be achieved when considering networks with overlapping memory states. In this case no additional conjunction units describing associative links in between two stored memories are needed. These associative links would be formed by other memories. Any new information learned by the network then acquires naturally meaningful associative links whenever it shares part of its constituent information with other memory states.

We do therefore consider a homogeneous network, with only one kind of constituent building block: the memories themselves. The memory states then show a dual functionality: depending on the initial condition, an associative link in between two or more activity centers could be either a stationary memory state by itself or it could serve to form an association in between two sequential memory states in the course of a thought process.

We present a generalized NN model capable of simulating the here defined kind of thought processes [Gro05]. We do not claim that actual thought processes in biological cybernetic systems (in our brain for instance) will be described accurately by this model. However, the kind of thought processes proposed here seem to be a mandatory requirement if a homogeneous associative network without an external regulative unit wants to acquire true information-processing capabilities. In such kind of networks the information-processing must be self-organized by an autonomous dynamical process. From a functional point of view it is evident that this self-regulated information processing needs to be implemented in biological cognitive systems, like the human brain, in one way or another.

We note that these self-organized association processes work only within dense associative networks, where essentially all activity centers are connected among themselves, forming what is called in network-theory a *giant strongly connected-component* [DM03]. In a sparse network there would be many unconnected sub-clusters incapable to communicate autonomously. Therefore we consider here dense and homogeneous associative networks (dHAN) and one might argue that the human brain does fall into this category.

Associative and Homogeneous Networks

We consider an associative network with N sites, which we also call activity centers (AC). Each AC represents some specific biologically relevant information, normally in a highly preprocessed form [HW65]. Examples are ACs for colors, shapes, distances, movements, sounds and so on. Each AC is characterized by the individual activity $x_i \in [0, 1]$ ($i = 1, \dots, N$). An AC is active when x_i is close to one. We identify ensembles of active ACs with memory states [MPR98].

In addition to the activity levels x_i we introduce for every AC a new variable $\varphi_i \in [0, 1]$, which characterizes the level of the individual activity reservoirs. This variable plays, as it will become clear from the discussions further below, a key role in facilitating the self-sustained thought process and distinguishes the dHAN from standard neural networks [MRS95].

We consider here a continuous-time evolution, $x_i = x_i(t)$ and $\varphi_i = \varphi_i(t)$. The following set of ODEs (with $i, j = 1, \dots, N$)

$$\dot{x}_i = (1 - x_i)\Theta(r_i)r_i + x_i[1 - \Theta(r_i)]r_i, \quad (2.130)$$

$$r_i = b_i + g(\varphi_i)w_{ij}x_j + z_{ij}x_jf(\varphi_j), \quad (2.131)$$

$$\dot{\varphi}_i = \Gamma_{\varphi}^+ \Theta(x_c - x_i)(1 - \varphi_i) - \Gamma_{\varphi}^- \Theta(x_i - x_c) \varphi_i, \quad (2.132)$$

determine the time-evolution of all $x_i(t)$. Here the r_i are growth rates and the b_i the respective biases [Gro05].²⁵

The function $\Theta(r)$ occurring in (2.130) is the *step function*:

$$\Theta(r) = \begin{cases} 1, & \text{for } r > 0, \\ 0, & \text{for } r \leq 0. \end{cases}$$

The dynamics, (2.130) and (2.132), respects the normalization $x_i \in [0, 1]$ and $\varphi_i \in [0, 1]$ due to the pre-factors $(1 - x_i)$, $(1 - \varphi_i)$ and x_i , φ_i for the growth and depletion processes.

²⁵ The differential equations (2.130) and (2.131) are akin to the Lotka-Volterra equations discussed by Fukai and Tanaka [FT97]. Also, the time-unit is arbitrary in principle and could be tuned, as well as most of the parameters entering (2.130) and (2.131), in order to reproduce neuro-biologically observed time-scales. For convenience one could take a millisecond for the time unit, or less.

The NN-type interactions in between the activity centers are given by the matrices $0 \leq w_{ij} \leq w$ and $z_{ij} \leq -|z|$ for excitatory and inhibitory connections respectively. The breakdown of the link-matrix in an excitatory and inhibitory sector can be considered as a reflection of the biological observation that excitatory and inhibitory signals are due to neurons and interneurons respectively. Any given connection is either excitatory or inhibitory, but not both at the same time,²⁶

$$w_{ij}z_{ij} \equiv 0, \quad \text{for all pairs } (i, j).$$

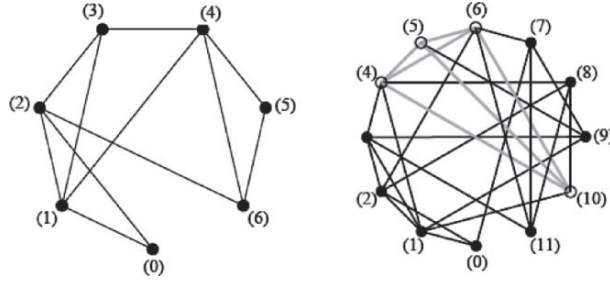


Fig. 2.49. Two small networks for illustration. Left: A 7-center network with five stable memory states: (0,1,2), (1,2,3), (1,3,4), (4,5,6) and (2,6). Right: A 12-center network with 7 2-center memory states, 7 3-center memory states and one 4-center memory state (which is highlighted). It contains a total of 28 links, which are non-zero matrix-elements of w_{ij} (modified and adapted from [Gro05]).

We consider here the recurrent case with $w_{ij} = w_{j,i}$ and $z_{ij} = z_{j,i}$, but the model works fine also when this symmetry is partially broken. This will happen anyhow dynamically via the reservoir-functions $f(\varphi)$ and $g(\varphi)$. These functions govern the interaction in between the activity levels x_i and the reservoir levels φ_i . They may be chosen as washed-out step functions of a sigmoidal form like

$$g(\varphi) = g^{(min)} + \left(1.0 - g^{(min)}\right) \frac{\text{atan}[(\varphi - \varphi_c^{(g)})/\Gamma_\varphi] - \text{atan}[(0 - \varphi_c^{(g)})/\Gamma_\varphi]}{\text{atan}[(1 - \varphi_c^{(g)})/\Gamma_\varphi] - \text{atan}[(0 - \varphi_c^{(g)})/\Gamma_\varphi]}, \quad (2.133)$$

with a suitable width Γ_φ . The effect of the reservoir functions depends on the value of the respective reservoir-levels φ_i , which are governed by (2.132).

For $x_i > x_c$ (high activity level) the reservoir-level φ_i decreases with the rate Γ_φ^- . For $x_i < x_c$ (low activity level) the reservoir-level φ_i increases with the rate Γ_φ^+ . A low reservoir level will have two effects: The ability to suppress another activity center via an inhibitory link z_{ij} , which will be reduced by

²⁶ We do not consider here self-interactions (i.e., auto-associations): $w_{i,i} = z_{i,i} \equiv 0$.

$f(\varphi_i) \in [0, 1]$ and the activation by other centers via an excitatory link w_{ij} , which will be reduced by $g(\varphi_i) \in [0, 1]$, see (2.131).

The dynamics induced by (2.130) leads to a relaxation towards the next stable memory state within a short time-scale of $\Gamma_r^{-1} \approx |w_{ij}|^{-1} \approx |z_{ij}|^{-1}$ (for the non-zero matrix-elements of the link-matrices). Choosing the rates $\Gamma_\varphi^\pm$ for the reservoir dynamics to be substantially smaller than the relaxation-rates Γ_r we get a separation of time-scales for the stabilization of memory states and for the depletion/filling of the activity reservoirs $\varphi_i(t)$ described by (2.132).

Memory States

We consider here memory states which contain only a finite number, typically between 2 and 7, of constituent activity centers. This is a key difference between the dHAN investigated here and standard neural networks, where a finite fraction of all neurons might be active simultaneously [Rei98].

The stabilization of memory states made up of clusters with a finite number $Z = 2, 3, \dots$ of activity centers is achieved by an inhibitory background of links,

$$z_{ij} \leq z < 0, \quad (\text{for all } w_{ij} = 0, i \neq j). \quad (2.134)$$

In Figure 2.49 above we illustrate a 7-center network, where the black lines are the excitatory links, i.e., the non-zero matrix-elements of w_{ij} . All pairs (i, j) of activity-centers not connected by a line have $z_{ij} \leq -|z|$. If $|z|$ is big enough, then only those clusters of activity centers are dynamically stable, in which all participating centers are mutually connected (see [Gro05]). To see why, we consider an AC (i) outside a Z -center memory state (MS). The site (i) cannot have links (finite w_{ij}) to all of the activity centers (j) making up this memory state. Otherwise (i) would be part of this MS. There are therefore maximally $Z - 1$ positive connections in between (i) and the MS. The dynamical stability of the memory state is guaranteed if the total link-strength between (i) and the MS is not too strong,

$$|z| > \sum_{j \in \text{MS}} w_{ij}, \quad |z| > (Z - 1)w, \quad (2.135)$$

where the second inequality holds for the uniform case, $w_{ij} \equiv w > 0$.

For an illustration of this relation we consider the case $x_3 = x_4 = x_5 = x_6 = 0$ and $x_0 = x_1 = x_2 = 1$ for the 7-center network of Figure 2.49. The growth-rate for center (3) is then: $r_3 = 2w - |z|$. For $2w - |z| > 0$ center (3) would start to become active and a spurious state (1,2,3,4) would result. Taking $2w - |z| < 0$ both (0,1,2) and (1,2,3) are stable and viable 3-center memory states.

A ‘spurious memory state’ of the network would occur if a group of ACs remains active for a prolonged time even though this grouping does not correspond to any stored memory state. No such spurious memory state is dynamically stable when (2.135) is fulfilled. For the simulation presented here

we have chosen $|z|/w > 6$. This implies that memory states with up to $Z = 7$ activity centers are stable, see (2.135).

This kind of encoding of the link-matrices is called a ‘winners-take-all’ situation²⁷, since fully interconnected clusters will stimulate each other via positive intra-cluster w_{ij} . There will be at least one negative z_{ij} -link in between an active center of the winning memory state to every out-of-cluster AC, suppressing in this way the competing activity of all out-of-cluster activity centers.

Hierarchical Memory States

In the above discussion we have considered in part the uniform case $w_{ij} \equiv w$ for all non-zero excitatory links. In this case, all features making up a memory state are bound together with the same strength. Such a memory state has no internal structure, it is just the reunion of a bunch of semantic nodes with no additional relations in between them. Memory states corresponding to biological relevant objects will however exhibit in general a hierarchical structure [RP99] [MF00]. Let us give an example: A memory state denoting a ‘boy’ may involve a grouping of ACs corresponding to (face), (shirt), (pants), (legs), (red), (green) and so on. This memory state is well defined in our model whenever there are positive links $w_{ij} > 0$ in between all of them.

There is now the need for additional information like: is ‘red’ the color of the shirt or of the trousers? That is, there is the need to ‘bind’ the color red preferentially to one of the two pieces of clothes. It is possible to encode this internal information into the memory state ‘boy’ (face, shirt, pants, legs, red, green,...) by appropriate modulation of the internal connections. In order to encode for instance that red is the color of the shirt, one sets the link (red)–(shirt) to be much stronger than the link (red)–(pants) or (red)–(legs). This is perfectly possible and in this way the binding of (red) to (shirt) is achieved. The structure of the memory states defined here for the dHAN is therefore flexible enough to allow for a (internal) hierarchical object representation.

No confusion regarding the colors of the shirt and of the pants arises in the above example when variable link-strengths w_{ij} are used. Note however, that this is possible only because small and negative links z_{ij} are not allowed in our model, a key difference to the most commonly used neural-network models. If weak inhibitory links would be present, the boundary of memory states could not be defined precisely. There would be no qualitative difference in between a small negative and a small positive synapsing strength. Furthermore, the stability condition (2.135) would break down.

²⁷ The winners-take-all setting here differs from the so-called ‘K-winners-take-all’ configuration in which the K most active neurons suppress the activities of all other neurons via an inhibitory background (see [KZ95]).

Thought Processes Dynamics

In Figure 2.50 we present an autonomous thought process within a 12-center network with 15 stable memory states, illustrated in Figure 2.49 above. We have chosen a small network here to discuss the properties of the dynamical thought process in detail. The model is however completely scalable and we have performed simulations of networks containing several thousands of sites without any problem on standard computers. The choice of synchronous or asynchronous updating procedures is arbitrary here, due to the continuous-time formulation. No particular care needs to be taken for the integration of the differential equations (2.130) and (2.132) as the dynamical process has relaxational properties for the short-time dynamics of both the activities $x_i(t)$ as well as for the reservoir levels $\varphi_i(t)$. The model is numerically robust. The dynamics is numerical stable also for the long-time evolution of the activities $x_i(t)$ which is driven by the respective reservoir levels via the reservoir-functions $f(\varphi)$ and $g(\varphi)$. The reservoir levels are depleted very slowly and the

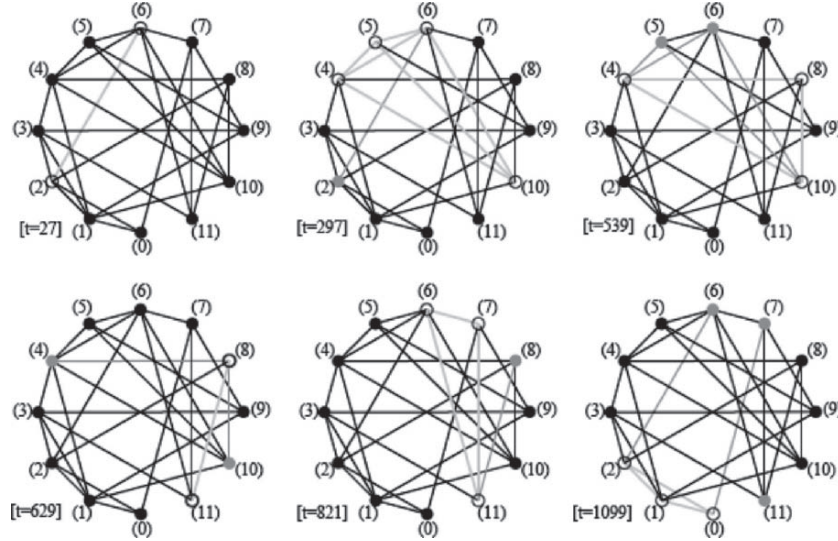


Fig. 2.50. The thought process $(2,6) \rightarrow (4,5,6,10) \rightarrow (4,8,10) \rightarrow (8,11) \rightarrow (6,7,11) \rightarrow (0,1,2)$ of a 12-site network with 7 2-center, 7 3-center and one 4-center memory state. The non-zero excitatory links $w_{ij} > 0$ differ from the uniform level w randomly by at most 5% (modified and adapted from [Gro05]).

memory-stable becomes unstable and a different memory-state takes over only after a substantial time has passed. Any two subsequent memory states are connected by one or more links. Excitatory links have a dual functionality: To stabilize a transient memory state and to associate one memory state

with a subsequent one. The model does therefore realize the two postulates for self-regulated associative thought processes set forth in the introduction.

Biologically speaking it is a ‘waste of time’ if individual memory states remain active for an exceedingly long interval. Note that the sequence of memory-state is identical for both parameter sets, the dynamics is stable over a wide range of parameter-values. The history of memory states goes through a cycle, as it is evident for the simulations using the parameter set (b), since the phase-space is finite. The 12-site cluster used in this simulation contains 15 different memory states and the cycle runs over 6 distinct states, a substantial fraction of the total. For larger networks with their very high numbers of stored memory-states [BDS89] the cycle length will be in general very long for any practical purposes. We have, however, not yet performed a systematic study of the cycle-length on the system properties along the same lines usually done for random boolean networks [Sch01].

Note that binary cycles do not occur for the parameters used here. We have chosen $\Gamma_{\varphi}^{+} < \Gamma_{\varphi}^{-}$ and the reservoir levels therefore take a while to fill-up again. Active memory states can therefore not reactivate their predecessors, which have necessarily low reservoir levels and small values for $g(\varphi)$. The same temporal asymmetry could be achieved by choosing $w_{ij} \neq w_{j,i}$. We do not rule-out the use of asymmetric link-matrices for the dHAN, but it is an important property of the dHAN to be able to establish a time direction autonomously. The types of memory-states storable in the dHAN would otherwise be limited to asymmetric states.

Biological Considerations

At first sight the model (2.130) possesses an non-biological feature. Neglecting the $f(\varphi)$ and $g(\varphi)$ for a moment, the total effective link strength $w_{ij} + z_{ij}$ is discontinuous: Either strongly negative ($w_{ij} = 0$, $z_{ij} \leq -|z|$), or weakly positive ($0 \leq w_{ij} \leq w$, $z_{ij} = 0$), as illustrated in Figure 2.51. This property of the link-matrices between the constituent activity centers is crucial for the whole model. Recall that it is essential for the stability of the individual memory states, and it forms the basis for hierarchical object representations. It constitutes a key difference between our and other models of neural networks [Sch01].

The effective link-strength $w_{ij} + z_{ij}$ does however not correspond to the bare synapsing-strength in biological neural assemblies. It represents an effective coupling in between local or distributed centers of neural activities and this kind of discontinuous behavior may actually result quite naturally from a simple coupling via intermediate inhibitory interneurons, as illustrated in Figure 2.51. When the interneuron is active, the effective coupling is strongly inhibitory. When the interneuron is quiet, the coupling is weakly excitatory with Hebbian-type learning. When the interneuron is active, it might as well inhibit the target neuron completely. Integrating out the intermediate inhibitory

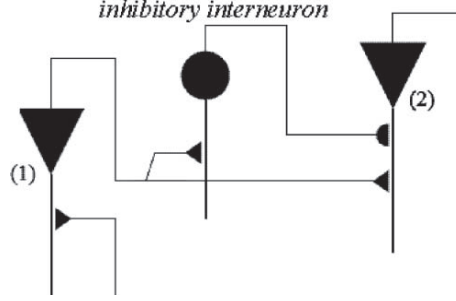


Fig. 2.51. a network of interacting neurons leading to an effective discontinuous interneural coupling: the excitatory neuron (1) connects both directly to neuron (2) and indirectly via the inhibitory interneuron. An activation of the interneuron by neuron-(1) may lead to complete inhibition of neuron-(2), masking completely the direct excitatory (1)–(2) link (modified and adapted from [Gro05]).

interneuron leads in this way to an effective discontinuous inter-neuron coupling.

Biologically speaking, one observes that inhibitory synapses are placed all over the target neurons, on the dendrites, the soma and on the axon itself. Excitatory synapses are, however, located mainly on the dendrites. This observation suggests that inhibitory synapses may indeed preempt the postsynaptic excitatory potentials, giving functionally rise to a model like the one proposed here. For more details see [Gro05].

2.5 Cellular and Evolutionary Approach

In this section we give a brief review on cellular automata (CA) and genetic algorithms (GA) approach to brain and cognition. Formally, here we are dealing with a *cellular evolution* 2-functor $\mathcal{E}$, given by

$$\begin{array}{ccc}
 \begin{array}{ccc}
 A & \xrightarrow{f} & B \\
 \downarrow h & \text{CURRENT} & \downarrow g \\
 & \text{CA/GA} & \\
 \downarrow & \text{STATE} & \\
 C & \xrightarrow{k} & D
 \end{array} & \xRightarrow{\mathcal{E}} & \begin{array}{ccc}
 \mathcal{E}(A) & \xrightarrow{\mathcal{E}(f)} & \mathcal{E}(B) \\
 \downarrow \mathcal{E}(h) & \text{DESIRED} & \downarrow \mathcal{E}(g) \\
 & \text{CA/GA} & \\
 \downarrow & \text{STATE} & \\
 \mathcal{E}(C) & \xrightarrow{\mathcal{E}(k)} & \mathcal{E}(D)
 \end{array}
 \end{array}$$

Here $\mathcal{E}$ represents an association/projection functor from the *source 2-category* of the current cellular state, defined as a commutative square of small cellular categories $A, B, C, D, \dots$ of *current cellular components* and their causal interrelations $f, g, h, k, \dots$, onto the *target 2-category* of the desired cellular state, defined as a commutative square of small cellular categories $\mathcal{E}(A), \mathcal{E}(B), \mathcal{E}(C), \mathcal{E}(D), \dots$.

$\mathcal{E}(C), \mathcal{E}(D), \dots$ of *evolved cellular components* and their causal interrelations $\mathcal{E}(f), \mathcal{E}(g), \mathcal{E}(h), \mathcal{E}(k)$. As in the previous section, each causal arrow in above diagram, e.g., $f : A \rightarrow B$, stands for a *generic cellular dynamorphism*.

2.5.1 Cellular Automata

In this subsection, following [Wol02, Wol84], we give an introduction to cellular automata. It is common in nature to find systems whose overall behavior is extremely complex, yet whose fundamental component parts are each very simple. The complexity is generated by the cooperative effect of many simple identical components. Much has been discovered about the nature of the components in physical and biological systems; little is known about the mechanisms by which these components act together to give the overall complexity observed. According to Steve Wolfram, what is needed is a general mathematical theory to describe the nature and generation of complexity.

Cellular automata (CA) are examples of mathematical systems constructed from many identical components, each simple, but together capable of complex behavior. From their analysis one may, on the one hand, develop specific models for particular systems, and, on the other hand, hope to abstract general principles applicable to a wide variety of complex systems.

1D Cellular Automata

Recall that a 1D CA consists of a line of sites, with each site carrying a value 0 or 1 (or in general $0, \dots, k-1$). The value α_i of the site at each position i is updated in discrete time steps according to an identical deterministic rule depending on a neighborhood of sites around it [Wol02, Wol84]:

$$\alpha_i^{t+1} = \varphi[\alpha_{i-r}^t, \alpha_{i-r+1}^t, \dots, \alpha_{i+r}^t]. \quad (2.136)$$

Even with $k = 2$ and $r = 1$ or 2 , the overall behavior of CA constructed in this simple way can be extremely complex.

Consider first the patterns generated by CA evolving from simple ‘seeds’ consisting of a few non-zero sites. Some local rules φ give rise to simple behavior; others produce complicated patterns. An extensive empirical study suggests that the patterns take on four qualitative forms (see Figure 2.52):

1. Disappears with time;
2. Evolves to a fixed finite size;
3. Grows indefinitely at a fixed speed; and
4. Grows and contracts irregularly.

Patterns of type 3 are often found to be self-similar or scale invariant. Parts of such patterns, when magnified, are indistinguishable from the whole. The patterns are characterized by a *fractal dimension*, with the most common

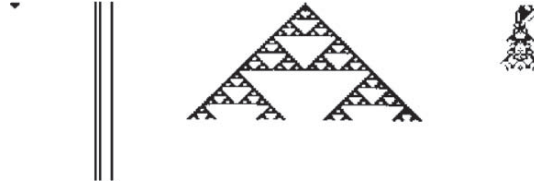


Fig. 2.52. Classes of patterns generated by the evolution of CA from simple ‘seeds’. Successive rows correspond to successive time steps in the CA evolution. Each site is updated at each time step according to equation (2.136) by CA rules that depend on the values of a neighborhood of sites at the previous time step. Sites with values 0 and 1 are represented by white and black squares, respectively. Despite the simplicity of their construction, patterns of some complexity are seen to be generated. The rules shown exemplify the four classes of behavior found. In the third case, a self-similar pattern is formed (adapted from [Wol02, Wol84]).

value $\log_2 3 \simeq 1.59$. Many of the self-similar patterns seen in natural systems may in fact, be generated by CA evolution.

Different initial states with a particular CA rule yield patterns that differ in detail, but are similar in form and statistical properties. Different CA rules yield very different patterns. An empirical study, nevertheless, suggests that four qualitative classes may be identified, yielding four characteristic limiting forms:

1. Spatially homogeneous state;
2. Sequence of simple stable or periodic structures;
3. Chaotic aperiodic behavior; and
4. Complicated localized structures, some propagating.

All CA within each class, regardless of the details of their construction and evolution rules, exhibit qualitatively similar behavior. Such universality should make general results on these classes applicable to a wide variety of systems modeled by CA.

CA Applications

Mathematical models of natural systems are usually based on differential equations which describe the smooth variation of one parameter as a function of a few others. Cellular automata provide alternative and in some respects complementary models, describing the discrete evolution of many (identical) components. Models based on CA are typically most appropriate in highly nonlinear regimes of physical systems, and in chemical and biological systems where discrete thresholds occur. Cellular automata are particularly suitable as models when *growth inhibition effects* are important [Wol02, Wol84].

As one example, CA provide global models for the growth of dendritic crystals (such as snowflakes). Starting from a simple seed, sites with values representing the solid phase are aggregated according to a 2D rule that accounts



Fig. 2.53. Evolution of small initial perturbations in CA, as shown by the difference (modulo two) between patterns generated from two disordered initial states differing in the value of a single site. The examples shown illustrate the four classes of behavior found. Information on changes in the initial state almost always propagates only a finite distance in the first two classes, but may propagate an arbitrary distance in the third and fourth classes (adapted from [Wol02, Wol84]).

for the inhibition of growth near newly-aggregated sites, resulting in a fractal pattern of growth. Nonlinear chemical reaction-diffusion systems give another example: a simple CA rule with growth inhibition captures the essential features of the usual partial differential equations, and reproduces the spatial patterns seen. Turbulent fluids may also potentially be modeled as CA with local interactions between discrete vortices on lattice sites [Wol02, Wol84].

If probabilistic noise is added to the time evolution rule (1), then CA may be identified as generalized *Ising-spin* models. Phase transitions may occur if retains some deterministic components, or in more than one dimension.

Cellular automata may serve as suitable models for a wide variety of biological systems. In particular, they may suggest mechanisms for biological pattern formation. For example, the patterns of pigmentation found on many mollusc shells bear a striking resemblance to patterns generated by class 2 and 3 CA, and CA models for the growth of some pigmentation patterns have been constructed [Wol02, Wol84].

Two Approaches to CA Mathematics

Rather than describing specific applications of CA, here we concentrate on general mathematical features of their behavior. Two complementary approaches provide characterizations of the four classes of behavior [Wol02, Wol84].

In the first approach, CA are viewed as discrete dynamical systems (see, e.g., [GH83]), or discrete idealizations of partial differential equations. The set of possible (infinite) configurations of a CA forms a *Cantor set*. CA evolution may be viewed as a continuous mapping on this Cantor set. Quantities such as entropies, dimensions and Lyapunov exponents may then be considered for CA.

In the second approach, CA are instead considered as information-processing systems (see, e.g., [HU79]), or parallel-processing computers of simple construction. Information represented by the initial configuration is

processed by the evolution of the CA. The results of this information processing may then be characterized in terms of the types of formal languages generated.²⁸

CA Entropies and Dimensions

Most CA rules have the important feature of irreversibility: several different configurations may evolve to a single configuration, and, with time, a contracting subset of all possible configurations appears. Starting from all possible initial configurations, the CA evolution may generate only special ‘organized’ configurations, and ‘self-organization’ may occur.

For class 1 CA, essentially all initial configurations evolve to a single final configuration, analogous to a limit point in a continuous dynamical system. Class 2 CA evolve to limit sets containing essentially only periodic configurations, analogous to limit cycles. Class 3 CA yield chaotic aperiodic limit sets, containing analogues of *strange attractors* [Wol02, Wol84].

Entropies and dimensions give a generalized measure of the density of the configurations generated by CA evolution. The (set) dimension or limiting (topological) entropy for a set of CA configurations is defined as (compare with [GH83]):

$$d^{(x)} = \lim_{X \rightarrow \infty} \frac{1}{X} \log_k N(X), \quad (2.137)$$

where $N(X)$ gives the number of distinct sequences of X -site values that appear. For the set of possible initial configurations, $d^{(x)} = 1$. For a limit set containing only a finite total number of configurations, $d^{(x)} = 0$. For most class 3 CA, $d^{(x)}$ decreases with time, giving, $0 < d^{(x)} < 1$, and suggesting that a fractal subset of all possible configurations occurs.

A dimension or limiting entropy $d^{(t)}$ corresponding to the time series of values of a single site may be defined in analogy with equation (2.137)²⁹ $d^{(t)} = 0$, for periodic sets of configurations.

Both $d^{(x)}$ and $d^{(t)}$ may be modified to account for the probabilities of configurations by defining

$$d_{\mu}^{(x)} = - \lim_{X \rightarrow \infty} \frac{1}{X} \sum_{i=1}^{k^{\mu}} p_i \log_k p_i, \quad (2.138)$$

and its $d^{(t)}$ -analogue, where p_i are probabilities for possible length X -sequences. These measure dimensions may be used to delineate the large time behavior of the different classes of CA:³⁰

²⁸ Note that the mechanisms for information processing in natural system appear to be much closer to those in CA than in conventional serial-processing computers: CA may, therefore, provide efficient media for practical simulations of many natural systems.

²⁹ The analogue of equation (2) for a sufficiently wide patch of sites yields a topologically-invariant entropy for the CA mapping.

³⁰ Dimensions are usually undefined for class 4 CA.

1. $d_\mu^{(x)} = d_\mu^{(t)} = 0$;
2. $d_\mu^{(x)} > 0, d_\mu^{(t)} = 0$;
3. $d_\mu^{(x)} > 0, d_\mu^{(t)} > 0$.

CA Information Propagation

Cellular automata may also be characterized by the stability or predictability of their behavior under small perturbations in initial configurations, usually resulting from a change in a single initial site value (see Figure 2.53). Such perturbations have characteristic effects on the four classes of CA:

1. No change in final state;
2. Changes only in a finite region;
3. Changes over an ever-increasing region; and
4. Irregular changes.

In class 1 and 2 CA, information associated with site values in the initial state propagates only a finite distance; in class 3 CA, it propagates an infinite distance at a fixed speed, while in class 4 CA, it propagates irregularly but over an infinite range. The speed of information propagation is related to the Lyapunov exponent for the CA evolution, and measures the degree of sensitivity to initial conditions. It leads to different degrees of predictability for the outcome of CA evolution [Wol02, Wol84]:

1. Entirely predictable, independent of initial state;
2. Local behavior predictable from local initial state;
3. Behavior depends on an ever-increasing initial region; and
4. Behavior effectively unpredictable.

Information propagation is particularly simple for the special class of additive CA (whose local rule function φ is linear modulo k), in which patterns generated from arbitrary initial states may be obtained by superposition of patterns generated by evolution of simple initial states containing a single non-zero site. A rather complete algebraic analysis of such CA may be given. Most CA are not additive; however, with special initial configurations it is often possible for them to behave just like additive rules. Thus, for example, the evolution of an initial configuration consisting of a sequence of 00 and 01 diagrams under one rule may be identical to the evolution of the corresponding 'blocked' configuration consisting of 0 and 1 under another rule. In this way, one rule may simulate another under a blocking transformation (analogous to a renormalization group transformation). Evolution from an arbitrary initial state may be attracted to (or repelled from) the special set of configurations for which such a simulation occurs. Often several phases exist, corresponding to different blocking transformations: sometimes phase boundaries move at constant speed, and one phase rapidly takes over; in other cases, phase boundaries execute random walks, annihilating in pairs, and leading to a slow

increase in the average domain size. Many rules appear to follow attractive simulation paths to additive rules, which correspond to fixed points of blocking transformations, and thus exhibit self similarity. The behavior of many rules at large times, and on large spatial scales, is therefore determined by the behavior of additive rules.

CA Thermodynamics

Decreases with time in the spatial entropies and dimensions of equations (2.137)–(2.138) signal irreversibility in CA evolution. Some CA rules are, however, reversible, so that each and every configuration has a unique predecessor in the evolution, and the spatial entropy and dimension of equations (2.137)–(2.138) remain constant with time.

Now, conventional *thermodynamics* gives a general description of systems whose microscopic evolution is reversible; it may, therefore, be applied to reversible CA. As usual, the ‘fine-grained’ entropy for sets (ensembles) of configurations, computed as in (2.138) with perfect knowledge of each site value, remains constant in time. The ‘coarse-grained’ entropy for configurations is, nevertheless, almost always non-decreasing with time, as required by the second law of thermodynamics. Coarse graining emulates the imprecision of practical measurements, and may be implemented by applying almost any contractive mapping to the configurations (a few iterations of an irreversible CA rule suffice). For example, coarse-grained entropy might be computed by applying (2.138) to every fifth site value. In an ensemble with low coarse-grained entropy, the values of every fifth site would be highly constrained, but arbitrary values for the intervening sites would be allowed. Then in the evolution of a class 3 or 4 CA the disorder of the intervening site values would ‘mix’ with the fifth-site values, and the coarse-grained entropy would tend towards its maximum value. Signs of self-organization in such systems must be sought in temporal correlations, often manifest in ‘fluctuations’ or meta-stable ‘pockets’ of order.

While all fundamental physical laws appear to be reversible, macroscopic systems often behave irreversibly, and are appropriately described by irreversible laws. Thus, for example, although the microscopic molecular dynamics of fluids is reversible, the relevant macroscopic velocity field obeys the irreversible *Navier–Stokes equations*. Conventional thermodynamics does not apply to such intrinsically irreversible systems; new general principles must be found. Thus, for CA with irreversible evolution rules, coarse-grained entropy typically increases for a short time, but then decreases to follow the fine grained entropy. Measures of the structure generated by self-organization in the large time limit are usually affected very little by coarse graining.

CA and Formal Language Theory

Quantities such as entropy and dimension, suggested by information theory, give only rough characterizations of CA behavior. Computation theory sug-

gests more complete descriptions of self-organization in CA (and other systems). Sets of CA configurations may be viewed as formal languages, consisting of sequences of symbols (site values) forming words according to definite grammatical rules.

The set of all possible initial configurations corresponds to a trivial formal language. The set of configurations obtained after any finite number of time steps are found to form a regular language. The words in a regular language correspond to the possible paths through a finite graph representing a finite state machine. It can be shown that a unique smallest finite graph reproduces any given regular language (see [HU79]). Examples of such graphs are shown in Figure 2.54. These graphs give complete specifications for sets of CA configurations (ignoring probabilities). The number of nodes in the smallest graph corresponding to a particular set of configurations may be defined as the 'regular language complexity' of the set. It specifies the size of the minimal description of the set in terms of regular languages. Larger correspond to more complicated sets.

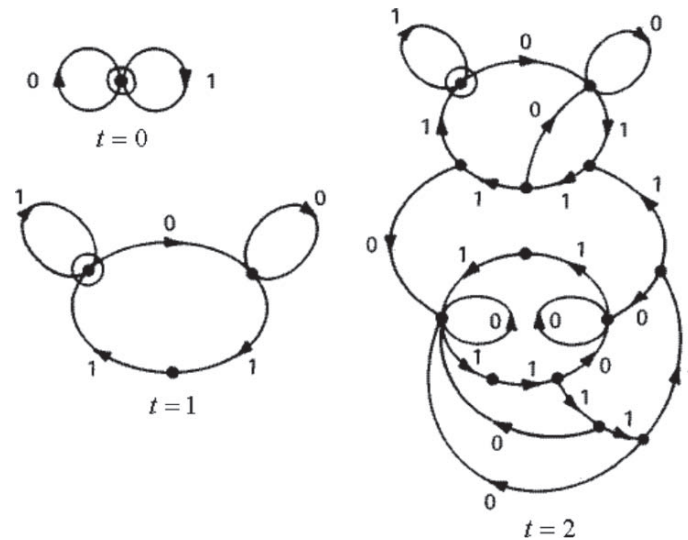


Fig. 2.54. Graphs representing the sets of configurations generated in the first few time steps of evolution according to a typical class 3 CA rule ($k = 2, r = 1$, rule number 126). Possible configurations correspond to possible paths through the graphs, beginning at the encircled node. At $t = 0$, all possible configurations are allowed. With time, a contracting subset of configurations are generated (e.g., after one time step no configuration containing the sequence of site value 101 can appear). At each time step, the complete set of possible configurations forms a regular formal language: the graph gives a minimal complete specification of it. The number of nodes in the graph gives a measure of the complexity Ξ of the set, viewed as a regular language. As for other class 3 CA, the complexity of the sets Ξ grows rapidly with time (modified and adapted from [Wol02, Wol84]).

The regular language complexity Ξ for sets generated by CA evolution almost always seems to be nondecreasing with time. Increasing Ξ signals increasing self-organization. Ξ may thus represent a fundamental property of self-organizing systems, complementary to entropy. It may, in principle, be extracted from experimental data [Wol02, Wol84].

Cellular automata that exhibit only class 1 and 2 behavior always appear to yield sets that correspond to regular languages in the large time limit. Class 3 and 4 behavior typically gives rise, however, to a rapid increase of Ξ with time, presumably leading to limiting sets not described by regular languages.

Formal languages are recognized or generated by idealized computers with a ‘central processing unit’ containing a fixed finite number of internal states, together with a ‘memory’. Four types of formal languages are conventionally identified, corresponding to four types of computer:

1. Regular languages: no memory required.
2. Context-free languages: memory arranged as a last-in, first-out stack.
3. Context-sensitive languages: memory as large as input word required.
4. Unrestricted languages: arbitrarily large memory required (general Turing machine).

Examples are known of CA whose limiting sets correspond to all four types of language. Arguments can be given that the limit sets for class 3 CA typically form context-sensitive languages, while those for class 4 CA correspond to unrestricted languages.³¹

CA and Computation Theory

While dynamical systems theory concepts suffice to define class 1, 2 and 3 CA, computation theory is apparently required for class 4 CA. Varied and complicated behavior, involving many different time scales is evident. Persistent structures are often generated. It seems that the structures supported by this and other class 4 CA rule may be combined to implement arbitrary information processing operations. Class 4 CA would then be capable of universal computation: with particular initial states, their evolution could implement any finite algorithm. A few percent of CA rules with $k > 2$ or $r > 1$ are found to exhibit class 4 behavior: all these would then, in fact, be capable of arbitrarily complicated behavior. This capability precludes a smooth infinite size limit for entropy or other quantities: as the size of CA considered increases, more and more complicated phenomena may appear [Wol02, Wol84].

CA evolution may be viewed as a computation. Effective prediction of the outcome of CA evolution requires a short-cut that allows a more efficient

³¹ While a minimal specification for any regular language may always be found, there is no finite procedure to obtain a minimal form for more complicated formal languages; no generalization of the regular language complexity may thus be given.

computation than the evolution itself. For class 1 and 2 CA, such short cuts are clearly possible: simple computations suffice to predict their complete future. The computational capabilities of class 3 and 4 CA may, however, be sufficiently great that, in general, they allow no short-cuts. The only effective way to determine their evolution from a given initial state would then be by explicit observation or simulation: no finite formulae for their general behavior could be given.³² Their infinite time limiting behavior could then not, in general, be determined by any finite computational process, and many of their limiting properties would be formally undecidable. Thus, for example, the ‘halting problem’ of determining whether a class 4 CA with a given finite initial configuration ever evolves to the null configuration would be undecidable. An explicit simulation could determine only whether halting occurred before some fixed time, and not whether it occurred after an arbitrarily long time.

For class 4 CA, the outcome of evolution from almost all initial configurations can probably be determined only by explicit simulation, while for class 3 CA this is the case for only a small fraction of initial states. Nevertheless, this possibility suggests that the occurrence of particular site value sequences in the infinite time limit is in general undecidable. The large time limit of the entropy for class 3 and 4 CA would then, in general, be non-computable: bounds on it could be given, but there could be no finite procedure to compute it to arbitrary precision.³³

Undecidability and intractability are common in problems of mathematics and computation. They may well afflict all but the simplest CA. One may speculate that they are widespread in natural systems, perhaps occurring almost whenever nonlinearity is present. No simple formulae for the behavior of many natural systems could then be given; the consequences of their evolution could be found effectively only by direct simulation or observation.

For more details on CA, complexity and computation, see [Wol02].

2.5.2 Evolutionary Algorithms

Recall that in the last few decades, there has been an increasing interest in algorithms based on the *Darwinian evolution principle* [Hol75]. The so-called *genetic algorithms* [Gol88, Mit96], *evolutionary programming* [Fog96, Bac96], *game-playing strategies* [Dav87] and *genetic programming* [Koz92] have found a wide field of applications in various areas where optimization plays an important role, such AI, financial markets, engineering, physics etc. In particular, genetic algorithms tackle even complex problems with surprising efficiency and robustness. For example, in physics they have been used in calculations that

³² If class 4 CA are indeed capable of universal computation, then the variety of their possible behavior would preclude general prediction, and make explicit observation or simulation necessary.

³³ This would be the case if the limit sets for class 3 and 4 CA formed at least context-sensitive languages.

involve from single Schrödinger particles in diverse potentials to astrophysical systems, running through lattice systems, spin glass models, molecules and clusters. The differences among the various evolutionary algorithms can be found not in the basic principles but in the details of the selection, reproduction and mutation procedures [SRV04].

In general, an *evolutionary algorithm* is based on three main statements:

- It is a process that works at the chromosomic level. Each individual is codified as a set of chromosomes.
- The process follows the Darwinian theory of evolution, say, the survival and reproduction of the fittest in a changing environment.
- The evolutionary process takes place at the reproduction stage. It is in this stage when mutation and crossover occurs. As a result, the progeny chromosomes can differ from their parents ones.

Starting from a guess initial population, an evolutionary algorithm basically generates consecutive generations (offprints). These are formed by a set of chromosomes, or character (genes) chains, which represent possible solutions to the problem under consideration. At each algorithm step, a fitness function is applied to the whole set of chromosomes of the corresponding generation in order to check the goodness of the codified solution. Then, according to their fitting capacity, couples of chromosomes, to which the crossover operator will be applied, are chosen. Also, at each step, a mutation operator is applied to a number of randomly chosen chromosomes.

The two most commonly used methods to randomly select the chromosomes are:

- (i) *The roulette wheel algorithm*. It consists in building a roulette, so that to each chromosome corresponds a circular sector proportional to its fitness.
- (ii) *The tournament method*. After shuffling the population, their chromosomes are made to compete among them in groups of a given size (generally in pairs). The winners will be those chromosomes with highest fitness. If we consider a binary tournament, say the competition is between pairs, the population must be shuffled twice. This technique guarantees copies of the best individual among the parents of the next generation.

After this selection, we proceed with the sexual reproduction or crossing of the chosen individuals. In this stage, the survivors exchange chromosomic material and the resulting chromosomes will codify the individuals of the next generation. The forms of sexual reproduction most commonly used are:

- (i) With one crossing point. This point is randomly chosen on the chain length, and all the chain portion between the crossing point and the chain end is exchanged.
- (ii) With two crossing points. The portion to be exchanged is in between two randomly chosen points.

For the algorithm implementation, the crossover normally has an assigned percentage that determines the frequency of its occurrence. This means that not all of the chromosomes will exchange material but some of them will pass

intact to the next generation. As a matter of fact, there is a technique, named elitism, in which the fittest individual along several generations does not cross with any of the other ones and keeps intact until an individual fitter than itself appears.

Besides the selection and crossover, there is another operation, mutation, that produces a change in one of the characters or genes of a randomly chosen chromosome. This operation allows to introduce new chromosomic material into the population. As for the crossover, the mutation is handled as a percentage that determines its occurrence frequency. This percentage is, generally, not greater than 5%, quite below the crossover percentage.

Once the selected chromosomes have been crossed and muted, we need some substitution method. Namely, we must choose, among those individuals, which ones will be substituted for the new progeny. Two main substitution ways are usually considered. In one of them, all modified parents are substituted for the generated new individuals. In this way an individual does never coexist with its parents. In the other one, only the worse fitted individuals of the whole population are substituted, thus allowing the coexistence among parents and progeny.

Since the answer to our problem is almost always unknown, we must establish some criterion to stop the algorithm. We can mention two such criteria [SRV04]:

- (i) the algorithm is run along a maximum number of generations; and
- (ii) the algorithm is ended when the population stabilization has been reached, i.e., when all, or most of, the individuals have the same fitness.

Genetic Algorithms

Recall that *genetic algorithms* (GAs) form a class of *algorithms* used to find approximate solutions to difficult-to-solve problems, inspired by and named after biological processes of *inheritance*, *mutation*, *natural selection*, and the genetic crossover that occurs when parents mate to produce offspring. Genetic algorithms are a particular class of *evolutionary algorithms*.

GAs deal with problems of optimizing. An example of such a problem can be neural net training, that is, selection of such weights that produce minimal possible errors. GAs are based on the method of *random search*. The main deficiency of random search is that we never know how much time it will take to find the solution. Such a waste of time when solving a problem can be avoided by using methods developed in biology. Methods are used that have been discovered during research on evolution and origin of species. As it is known, only the most adjusted individuals survive in the process of evolution. Due to this the *population's fitness* grows, thus allowing it to survive in changeable environment.

Such an algorithm was first proposed by John Holland in the Michigan University in 1975 [Hol75]. It was called 'Holland's reproductive plan' and

laid the groundwork for practically every variant of genetic algorithms. Holland was the pioneering founder of much of today's work in genetic algorithms, which has moved on from a purely theoretical subject though based on computer modeling, to provide methods which can be used to solve some difficult problems today. Problems which appear to be particularly appropriate for solution by genetic algorithms include timetabling and scheduling problems, and many scheduling software packages are based on GAs.

The problem to be solved is represented by a list of parameters which can be used to drive an evaluation procedure. The list is evaluated, and a value of *goodness* or *fitness* is returned (see [Gol88, Mit96]).

Initially several such parameter lists are generated randomly, to form an initial pool of possible solutions. This is called the first generation pool.

All of the lists are evaluated, and effectively the pool is sorted with those having better fitness at the top, representing better solutions to the problem. Notice that *better* in this context is relative, as initial solutions are likely to be rather poor.

The next step of the algorithm is to generate a second generation pool of parameter lists, which is done using the *genetic operators selection, crossover* (or *recombination*), and *mutation*.

The first step in the construction of the next generation is to *select* a pair of lists for crossover. Selection is biased towards elements of the initial generation which have better fitness, though it is not so biased that poorer elements have no chance to participate. This can be done using *roulette wheel selection* or using a ranking method.

Then perform the *crossover* (or *recombination*) genetic operator. This results in a new pair of lists, which are added to the second generation pool. This can be repeated until there are an appropriate number of new lists in the second generation pool.

The next step is to mutate the newly developed pool, again by a process of selection, this time of individual lists, followed by application of the *mutation* genetic operator. This process results in a second generation pool of lists which is different from the initial generation, which is then evaluated and the fitness values for each list is obtained. Generally the average degree of fitness will have increased by this procedure for the second generation pool.

A slight variant of this method of pool generation is to allow some of the better lists from the first generation to carry over to the second. This form of genetic algorithm is known as *elite*.

The process continues by generating 3rd, 4th, 5th ... generations, until one of the generations contains solutions which are good enough.

There are several observations to make about the generation of solutions.

GAs may have a tendency to converge towards *local solutions* rather than *global solutions* to the problem to be solved as time goes on each generation will tend to have multiple copies of successful parameter lists, which require evaluation, and this can slow down the processing the most important genetic

operators are selection and crossover. Mutation is only necessary to ensure that potential solutions are not lost.

It is also important to note that there are several different variants of the basic GA. The simplest algorithm represents each parameter list as a bit string. Typically numeric parameters can be represented by integers, though it is possible to use floating point representations. The basic algorithm performs crossover and mutation at the bit level. Other variants treat the parameter list as lists of numbers, and crossover and mutation are performed so as to respect number boundaries.

Genetic algorithms are known to produce good results for some problems. Their major disadvantage is that they are relatively slow, compared to other methods, such as *random optimization*. Recent speed improvements have focused on speciation, wherein cross-over can only occur if individuals are closely-enough related.

On the other hand, *genetic programming* is a technique developed by John Koza [Koz92], which is similar, but in which computer programs are modified, and evaluated in the optimization process. Genetic programming algorithms typically require running time which is orders of magnitude greater than a GA, but may be able to solve some problems which GAs cannot do easily. Genetic programming also uses internal data structures which are based on the use of tree structures to represent the computer programs for adaptation, rather than list structures which are used in genetic algorithms.

Representation of Objects in GAs

It is well-known in biology that any organism can be represented by its *phenotype*, which virtually determines *what* exactly the object is in the real world, and its *genotype* containing all the information about the object at the chromosome set level. Each gene, that is the genotype's information element, is reflected in the phenotype. Thus, to be able to solve problems we have to represent every attribute of an object in a form suitable for use in genetic algorithms. All further operation of genetic algorithm is done on the genotype level, making the information about the object's internal structure redundant. This is why this algorithm is widely used to solve all sorts of problems.

In the most frequently used variant of genetic algorithm, an object's genotype is represented by bit strings. Each attribute of an object in the phenotype has a single corresponding gene in the genotype. The gene is represented by a bit string, usually of a fixed length, which represents the value of the attribute.

Encoding Attributes Represented by Integers

The simplest variant can be used to encode such attributes that is the bit value of the attribute. Then it will be quite easy to use a gene of certain length, sufficient to represent all possible values of such an attribute. Unfortunately this encoding method is not perfect. Its main disadvantage is that neighboring numbers differ in several bits' values. Thus, for example, such numbers as 7

and 8 in the bit representation have four different bits, which complicates the gene algorithm functioning and increases time necessary for its convergence. To avoid this problem another encoding method should be used, in which neighboring numbers have less differences, ideally differing in only one bit. One of such codes is the Gray code, which is appropriate to be used with genetic algorithms. The table below shows the Gray code values:

Binary coding			Coding using the Gray code		
Dec.code	Bin.value	Hex.value	Dec.code	Bin.value	Hex.value
0	0000	0h	0	0000	0h
1	0001	1h	1	0001	1h
2	0010	2h	3	0011	3h
3	0011	3h	2	0010	2h
4	0100	4h	6	0110	6h
5	0101	5h	7	0111	7h
6	0110	6h	5	0101	5h
7	0111	7h	4	0100	4h
8	1000	8h	12	1100	Ch
9	1001	9h	13	1101	Dh
10	1010	Ah	15	1111	Fh
11	1011	Bh	14	1110	Eh
12	1100	Ch	10	1010	Ah
13	1101	Dh	11	1011	Bh
14	1110	Eh	9	1001	9h
15	1111	Fh	8	1000	8h

Table 2.2. Correspondence between decimal codes and the Gray codes.

Accordingly, when encoding an integer-valued attribute, we break it into quadruples and then convert each quadruple according to *Gray code*.

Usually, there is no need to convert attribute values into gene values in practical use of GAs. In practice, inverse problem occurs, when it is necessary to find the attribute value from the corresponding gene value.

Thus, the problem of decoding gene values, which have corresponding integer-valued attributes, is trivial.

Encoding Attributes Represented by Floating-Point Numbers

The simplest coding method, which first comes to mind, is to use bit representation. However, this variant is equally imperfect as in the case of integers. For this reason, the following sequence is used in practice:

1. All the interval of the attribute's allowed values is split into segments with adequate accuracy.
2. The value of the gene is accepted as an integer defining the interval number (using the Gray code).

3. The midpoint number of the interval is taken as the parameter value.

Let us consider a specific example of the sequence of operations described above: Assume that the attribute values are located in the interval $[0, 1]$. During the encoding the segment is split into 256 intervals. Thus we will need 8 bits to code their numbers. Let us suppose the number of the gene is 00100101*bG* (the capital letter ‘*G*’ stands for ‘Gray code’). For a start we shall find the corresponding interval number using the following Gray code: $25hG \rightarrow 36h \rightarrow 54d$. Now let us see what interval corresponds to it. . . Simple calculation gives us the interval: $[0.20703125, 0.2109375]$. Then, the value of the parameter is $(0.20703125 + 0.2109375)/2 = 0.208984375$.

Encoding Nonnumeric Data

To encode nonnumeric data, we have to convert it into numbers. More detailed description can be found on our web site in the articles dedicated to the use of neural nets.

Finding an Object’s Phenotype from its Genotype

Thus, to find an object’s *phenotype* (i.e., values of the attributes describing the object) we only have to know the values of the genes corresponding to these attributes, i.e., the object’s *genotype*. The aggregate of the genes describing the object’s genotype represents the *chromosome*. In some implementations it is called an individual. Thus, when implementing genetic algorithm, a chromosome is a bit string of a fixed length. Each segment of a string has its corresponding gene. Genes inside a chromosome can have equal or different lengths. Genes of equal length are used most often. Let us consider an example of a chromosome and interpretation of its value. Let us assume that the object has five attributes, each encoded by a gene 4 elements long. Then, the length of the chromosome is $5 \cdot 4 = 20$ bits:

0010	1010	1001	0100	1101

Now we can define the values of the attributes:

Attribute	Gene value	Binary value of the attribute	Decimal value of the attribute
Attribute 1	0010	0011	3
Attribute 2	1010	1100	12
Attribute 3	1001	1110	14
Attribute 4	0100	0111	7
Attribute 5	1101	1001	9

Basic Genetic Operators

As it is known in the evolution theory, the way the parents' attributes are inherited by their offsprings is of high importance. In genetic algorithms an operator called *crossing* (also known as crossover or crossing over) is in charge of passing the attributes from parents to their offsprings. It works in the following way:

1. Two individuals are selected from the population to become parents;
2. A break point is determined (usually at random); and
3. The offspring is determined as concatenation of the first and the second parents' parts.

Let us see how this operator works:

Chromosome_1:	0000000000
Chromosome_2:	1111111111

Now, if we put the break after the third bit of the chromosome, then we have:

Chromosome_1:	0000000000	>>	000	1111111	Resulting_chromosome_1
Chromosome_2:	1111111111	>>	111	0000000	Resulting_chromosome_2

After that, one of the resulting chromosomes is taken as an offspring with the 0.5 probability.

The next genetic operator is intended for maintaining the diversity of individuals in the population. It is called *mutation*. When it is used on a chromosome, each bit in it gets inverted with a certain probability.

Besides, one more operator is used, called *inversion*. Applying it makes a chromosome break in two parts, which then trade places. This can be shown schematically as follows:

000	1111111	>>	1111111	000

Theoretically, these two genetic operators are enough to make the genetic algorithm work. However, in practice some additional operators are used, as well as modifications of these two operators. For instance, in addition to the single-point crossover (described above) there can be a multipoint one, when several break points (usually two) are formed. Besides, in some implementations of the algorithm the mutation operator performs the inversion of only one randomly selected bit of a chromosome.

The Basic Algorithm

Having found out how to interpret the values of the genes, we proceed to describing the genetic algorithm operation. Let us consider the flow chart of genetic algorithm operation in its classic variant.

1. Initialize the start time $t = 0$. At random fashion form the initial population consisting of k individuals: $B_0 = \{A_1, A_2, \dots, A_k\}$.
2. Calculate the *fitness* of every individual: $F_{Ai} = fit(A_i)$, ($i = 1 \dots k$), and of the population as a whole: $F_t = fit(B_t)$. The value of this function determines how suitable for solving the problem the individual described by this chromosome is.
3. Select the individual A_c from the population: $A_c = Get(B_t)$.
4. With a certain crossover probability P_c select the second individual from the population: $A_{c1} = Get(B_t)$, and apply the crossover operator: $A_c = Crossing(A_c, A_{c1})$.
5. With a certain mutation probability P_m apply the mutation operator: $A_c = mutation(A_c)$.
6. With a certain inversion probability P_i apply the inversion operator: $A_c = inversion(A_c)$.
7. Place the resulting chromosome in the new population: $insert(B_{t+1}, A_c)$.
8. Repeat steps 3 to 7 k times.
9. Increase the current epoch number $t = t + 1$.
10. If the stop condition is met, terminate the loop, else go to step 2.

Now let us examine in detail the individual steps of the algorithm. The steps 3 and 4 play the most important role in the successful operation of the algorithm when parent chromosomes are selected. Various alternatives are possible. The most frequently used *selection method* is called *roulette*. When using it, the probability of a chromosome selection is determined by its fitness, i.e., $P_{Get(A_i)} \sim Fit(A_i)/Fit(B_t)$. This method increases the probability of the attributes propagation that belong to the most adjusted individuals. Another frequently used method is the *tournament selection*. It means that several individuals (usually two) are selected in the population at random. The one wins which is more adjusted. Besides, in some implementations of the algorithm the so-called *elitism strategy* is used, which means that the best-adjusted individuals are guaranteed to enter the new population. Using the elitism method is usually helpful to accelerate the genetic algorithm convergence. The disadvantage of this strategy is increased probability of the algorithm getting in the local minimum.

Another important point is the algorithm stop criteria determination. Usually the highest limit of the algorithm functioning epochs is taken as such, or the algorithm is stopped upon stabilization of its convergence, normally measured by means of comparing the population's fitness on various epochs.

2.6 Fuzzy–Logic Approach

In this section we give a brief review on fuzzy–systems approach to brain and cognition. Formally, here we are dealing with a *fuzzy–system evolution* 2–functor $\mathcal{E}$, given by

$$\begin{array}{ccc}
 \begin{array}{ccc}
 A & \xrightarrow{f} & B \\
 \downarrow h & \text{CURRENT} & \downarrow g \\
 & \text{FUZZY} & \\
 & \text{STATE} & \\
 C & \xrightarrow{k} & D
 \end{array}
 & \xrightarrow{\mathcal{E}} &
 \begin{array}{ccc}
 \mathcal{E}(A) & \xrightarrow{\mathcal{E}(f)} & \mathcal{E}(B) \\
 \downarrow \mathcal{E}(h) & \text{DESIRED} & \downarrow \mathcal{E}(g) \\
 & \text{FUZZY} & \\
 & \text{STATE} & \\
 \mathcal{E}(C) & \xrightarrow{\mathcal{E}(k)} & \mathcal{E}(D)
 \end{array}
 \end{array}$$

Here $\mathcal{E}$ represents an association/projection functor from the *source 2–category* of the current fuzzy–system state, defined as a commutative square of small fuzzy–system categories $A, B, C, D, \dots$ of *current fuzzy–system components* and their causal interrelations $f, g, h, k, \dots$, onto the *target 2–category* of the desired fuzzy–system state, defined as a commutative square of small fuzzy–system categories $\mathcal{E}(A), \mathcal{E}(B), \mathcal{E}(C), \mathcal{E}(D), \dots$ of *evolved fuzzy–system components* and their causal interrelations $\mathcal{E}(f), \mathcal{E}(g), \mathcal{E}(h), \mathcal{E}(k)$. As in the previous section, each causal arrow in above diagram, e.g., $f : A \rightarrow B$, stands for a *generic fuzzy–system dynamorphism*.

To re–phrase Bart Kosko from ‘Fuzzy Future’, a neural–network acts like the sensory part of our brain: it learns fuzzy patterns; on the other hand, a fuzzy–logic system processes these fuzzy patterns and acts like the cognitive part of a brain: it reasons with patterns.

In this section we present the basics of *fuzzy logic* and related *fuzzy expert systems*. Fuzzy logic is a natural companion of ANNs.³⁴ It was initiated in 1965 by Lotfi Zadeh (see [Zad65, Zad78, Yag87]), professor of computer science at the University of California in Berkeley, as a way of processing data by allowing partial set membership rather than crisp set membership or non–membership. This approach to set theory was not really applied to control systems until the 90s due to insufficient computer capability prior to that time. Professor Zadeh reasoned that people did not require precise, numerical information input, and yet they were capable of highly adaptive control. If feedback controllers could be programmed to accept noisy, imprecise input, they would be much more effective and perhaps easier to implement.

2.6.1 The Concept of Fuzziness

‘Fuzzy Thinking’

‘There is no logic in logic’, pronounced the father of fuzzy logic, Lotfi Zadeh. His cryptic play–on–words, he explained, means that the kind of logic that

³⁴ In the previous decades quite common were the joint conferences for neural networks and fuzzy expert systems.

people use to solve most real world problems rather than the artificial problems for which mathematical solutions are available is not the kind of logic that engineers are taught in school. ‘An engineer can solve problems throughout his whole career without ever needing to resort to the brand of logic he was trained in’, said Zadeh. ‘Why? Because all people, even engineers, compute with words not the logical symbols taught in school’, Zadeh maintained. ‘In the future, computing will be done with words from natural languages, rather than with symbols that are far removed from daily life.’

In 1973, Zadeh proposed the concept of linguistic or fuzzy variables [Zad65, Zad78, Yag87]. Think of them as linguistic objects or words, rather than numbers. The sensor input is a noun, e.g., temperature, displacement, velocity, or pressure, etc. Since error is just the difference, it can be thought of the same way. The fuzzy variables themselves are adjectives that modify the variable (e.g., large positive error, small positive error, zero error, small negative error, and large negative error). As a minimum, one could simply have positive, zero, and negative variables for each of the parameters.

Additional ranges such as very large and very small could also be added to extend the responsiveness to exceptional or very nonlinear conditions, but are not necessary in a basic system. Normal logic is just not up to modeling the real world, claims Bart Kosko [Kos92, Kos93, Kos96, Kos99], perhaps the world’s most active proponent of fuzzy logic. According to Kosko, there is always *ambiguity* in our perceptions and measurements that is difficult to reflect in traditional logic. Probability attempts to reflect ambiguity by resorting to statistical averages over many events. But fuzzy theory describes the ambiguity of individual events. It measures the degree to which an event occurs, not whether it occurs.

Fuzzy Sets

Recall that a crisp (ordinary mathematical) set X is defined by a binary characteristic function $\chi_X(x)$ of its elements x

$$\chi_X(x) = \begin{cases} 1, & \text{if } x \in X, \\ 0, & \text{if } x \notin X, \end{cases}$$

while a fuzzy set is defined by a continuous characteristic function

$$\chi_X(x) = [0, 1],$$

including all (possible) real values between the two crisp extremes 1 and 0, and including them as special cases.

More precisely, a fuzzy set X is defined as a collection of ordered pairs

$$X = \{(x, \mu(x))\}, \quad (2.139)$$

where $\mu(x)$ is the *fuzzy membership function* representing the grade of membership of the element x in the set X . A single pair is called a *fuzzy singleton*.

Lotfi Zadeh claimed that many *sets* in the world that surrounds us are defined by a non-distinct boundary. Indeed, the *set of high mountains* is an example of such sets. Zadeh decided to extend two-valued logic, defined by the binary pair $\{0, 1\}$ to the whole continuous interval $[0, 1]$ thereby introducing a gradual transition from falsehood to truth. The original and pioneering papers on fuzzy sets by Zadeh [Zad65, Zad78, Yag87] explain the theory of fuzzy sets that result from the extension as well as a fuzzy logic based on the set theory.

Fuzzy sets are a further development of the mathematical concept of a set. Sets were first studied formally by German mathematician Georg Cantor (1845–1918). His theory of sets met much resistance during his lifetime, but nowadays most mathematicians believe it is possible to express most, if not all, of mathematics in the language of set theory. Many researchers are looking at the consequences of ‘fuzzifying’ set theory, and much mathematical literature is the result.

Conventional sets. A set is any collection of objects which can be treated as a whole. Cantor described a set by its members, such that an item from a given universe is either a member or not. Almost anything called a *set* in ordinary conversation is an acceptable set in the mathematical sense. A set can be specified by its members, they characterize a set completely. The list of members $A = \{0, 1, 2, 3\}$ specifies a finite set. Nobody can list all elements of an *infinite set*, we must instead state some property which characterizes the elements in the set, for instance the predicate $x > 10$. That set is defined by the elements of the *universe of discourse* which make the predicate true. So there are two ways to describe a set: explicitly in a list or implicitly with a predicate.

Fuzzy sets. Following Zadeh many sets have more than an *Either–Or* criterion for membership. Take for example the set of *young people*. A one year old baby will clearly be a member of the set, and a 100 years old person will not be a member of this set, but what about people at the age of 20, 30, or 40 years? Another example is a weather report regarding high temperatures, strong winds, or nice days. In other cases a criterion appears nonfuzzy, but is perceived as fuzzy: a speed limit of 60 kilometers per hour, a check-out time at 12 noon in a hotel, a 50 years old man. Zadeh proposed a *grade of membership*, such that the transition from membership to non-membership is gradual rather than abrupt.

The grade of membership for all its members thus describes a fuzzy set. An item’s grade of membership is normally a real number between 0 and 1, often denoted by the Greek letter μ . The higher the number, the higher the membership. Zadeh regards Cantor’s set as a special case where elements have full membership, i.e., $\mu = 1$. He nevertheless called Cantor’s sets *nonfuzzy*; today the term *crisp* set is used, which avoids that little dilemma.

The membership for a 50 year old in the set *young* depends on one's own view. The grade of membership is a precise, but subjective measure that depends on the context.

A fuzzy membership function is different from a statistical probability distribution. A possible event does not imply that it is probable. However, if it is probable it must also be possible. We might view a fuzzy membership function as our personal distribution, in contrast with a statistical distribution based on observations.

Universe of discourse. Elements of a fuzzy set are taken from a *universe of discourse*. It contains all elements that can come into consideration. Even the universe of discourse depends on the context. An application of the universe is to suppress faulty measurement data. In case we are dealing with a non-numerical quantity, for instance *taste*, which cannot be measured against a numerical scale, we cannot use a numerical universe. The elements are then said to be taken from a *psychological continuum*.

Membership Functions. Every element in the universe of discourse is a member of the fuzzy set to some grade, maybe even zero. The set of elements that have a non-zero membership is called the *support* of the fuzzy set. The function that ties a number to each element x of the universe is called the *membership function*.

Continuous and discrete representations. There are two alternative ways to represent a membership function in a computer: continuous or discrete. In the continuous form the membership function is a mathematical function, possibly a program. A membership function is for example bell-shaped (also called a π -curve), s -shaped (called an s -curve), a reverse s -curve (called z -curve), triangular, or trapezoidal. In the discrete form the membership function and the universe are discrete points in a list (vector). Sometimes it can be more convenient with a sampled (discrete) representation. As a very crude rule of thumb, the continuous form is more CPU intensive, but less storage demanding than the discrete form.

Normalization. A fuzzy set is *normalized* if its largest membership value equals 1. We normalize by dividing each membership value by the largest membership in the set, $a/\max(a)$.

Singletons. Strictly speaking, a fuzzy set A is a collection of ordered pairs: $A = \{(x, \mu(x))\}$.

Item x belongs to the universe and $\mu(x)$ is its *grade of membership* in A . A single pair $(x, \mu(x))$ is called a fuzzy *singleton*; thus the whole set can be viewed as the union of its constituent singletons.

Linguistic variables. Just like an algebraic variable takes numbers as values, a *linguistic variable* takes words or sentences as values [Yag87, Kos92].

The set of values that it can take is called its *term set*. Each value in the term set is a *fuzzy variable* defined over a *base variable*. The base variable defines the universe of discourse for all the fuzzy variables in the term set. In short, the hierarchy is as follows:

$$\text{linguistic variable} \rightarrow \text{fuzzy variable} \rightarrow \text{base variable}.$$

Primary terms. A *primary term* is a term or a set that must be defined a priori, for example *Young* and *Old*, whereas the sets *Very Young* and *Not Young* are modified sets.

Fuzzy set operations. A *fuzzy set operation* creates a new set from one or several given sets.

Let A and B be fuzzy sets on a mutual universe of discourse X . If these were ordinary (crisp) sets, we would have the following definitions:

The *intersection* of A and B is: $A \cap B \equiv \min\{A, B\}$, where *min* is an item-by-item minimum operation.

The *union* of A and B is: $A \cup B \equiv \max\{A, B\}$, where *max* is an item-by-item maximum operation.

The *complement* of A is: $\bar{A} \equiv 1 - A$, where in a each membership value is subtracted from 1.

However, as A and B are fuzzy sets, the following definitions are more appropriate:

The *intersection* of A and B is: $A \cap B \equiv \min\{\mu_A(X), \mu_B(X)\}$, where *min* is an item-by-item minimum operation.

The *union* of A and B is: $A \cup B \equiv \max\{\mu_A(X), \mu_B(X)\}$, where *max* is an item-by-item maximum operation.

The *complement* of A is: $\bar{A} \equiv 1 - \mu_A(X)$, where in a each membership value is subtracted from 1.

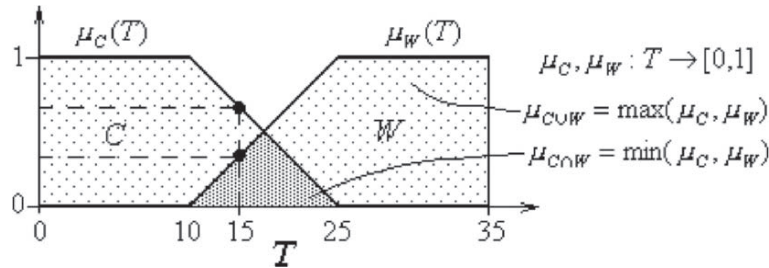


Fig. 2.55. Fuzzy-set description of *cold* (C) and *warm* (W) temperature (T), using the membership functions $\mu_C(T)$ and $\mu_W(T)$, respectively. For example, fuzzy answers to the questions “How cold is 15° ?” and “How warm is 15° ?” are given by: “ 15° is quite cold as $\mu_C(15) = 2/3$ ” and “ 15° is not really warm as $\mu_W(15) = 1/3$ ”, respectively.

Fuzzy Example

Using fuzzy membership functions $\mu(x)$, we can express both physical and non-physical quantities (e.g., temperature, see Figure 2.55) using *linguistic variables*.

Various logical combinations of such linguistic variables leads to the concept of fuzzy-logic control. Recall that basic logical operations AND, OR, NOT are defined as:

$$\begin{aligned} AND : C \cap W & - \text{ intersection of crisp sets } C, W, \\ OR : C \cup W & - \text{ union of crisp sets } C, W, \\ NOT : \bar{C} & - \text{ complement of a crisp set } C. \end{aligned}$$

The corresponding fuzzy-logic operations are defined as:

$$\begin{aligned} AND : \mu_{C \cap W}(T) &= \min\{\mu_C(T), \mu_W(T)\}, \\ OR : \mu_{C \cup W}(T) &= \max\{\mu_C(T), \mu_W(T)\}, \\ NOT : \mu_{\bar{C}}(T) &= 1 - \mu_C(T). \end{aligned}$$

Fuzziness of the Real World

The real world consists of all subsets of the universe and the only subsets that are not fuzzy are the constructs of classical mathematics.

From small errors to satisfied customers to safe investments to noisy signals to charged particles, each element of the real world is in some measure fuzzy. For instance, satisfied customers can be somewhat unsatisfied, safe investments somewhat unsafe and so on. What is worse, most events more or less smoothly transition into their opposites, making classification difficult near the midpoint of the transition. Unfortunately, textbook events and their opposites are crisp, unlike the real world. Take the proposition that there is a 50% chance that an apple is in the refrigerator. That is an assertion of crisp logic. But suppose upon investigation it is found that there is half an apple in the refrigerator, that is fuzzy.

But regardless of the realities, the crisp logic in vogue today assumes that the world is really unambiguous and that the only uncertainty is the result of random samples from large sets. As the facts about these large sets become better known, the randomness supposedly dissipates, so that if science had access to all the facts, it would disappear. Unfortunately, if all the facts were in, a platypus would remain only roughly an mammal.

On the other hand, fuzzy logic holds that uncertainty is deterministic and does not dissipate as more elements of a set are examined. Take an ellipse, for instance. It is approximately a circle, to whatever degree that it resembles a perfect circle. There is nothing random about it. No matter how precisely it is measured it remains only approximately a circle. All the facts are in and yet uncertainty remains.

Traditional crisp logic has a difficult time applying itself to very large sets, since probability fades to unity, as well as to individual events where probabilities cannot be defined at all. Nevertheless, crisp logic continues to rein supreme based on long standing western traditions that maintain that rationality would vanish if there were not crisp logical ideals to which we should aspire. These laws of (rational) thought were first characterized by Aristotle as the principle of non-contradiction and the principle of the excluded middle. The principle of non-contradiction, stated in words, says that nothing can be both A and $\bar{A}$. The law of the excluded middle says that anything must be either A or $\bar{A}$.

‘Fuzziness is the denial of both these so-called laws’, says E. Cox [Cox92, Cox94]). The classical example is of a platypus which both is and is not a mammal. In such individual cases, even appending probability theory to crisp logic cannot resolve the paradox. For instance, take the now classical paradox formulated by B. Russell: If a barber shaves everyone in a village who does not shave himself, then who shaves the barber? This paradox was devised to assault G. Cantor’s set theory as the foundation for G. Boole’s digital logic. It has been restated in many forms, such as the liar from Crete who said that all Cretans are liars. Russell solved it by merely disqualifying such self-referential statements in his set theory. Probability theory solves it by assuming a population of barbers 50% of whom do, and 50% of whom do not, shave themselves. But fuzzy logic solves it by assigning to this individual barber a 50% membership value in the set self-shaving barbers. Further, it shows that there is a whole spectrum of other situations that are less fuzzy and which correspond to other degrees of set membership. Such as, barbers who shave themselves 70% of the time.

Kosko illustrates these various degrees of ambiguity by geometrically plotting various degrees of set membership inside a *unit fuzzy hypercube* $[0, 1]^n$ [Kos92, Kos93, Kos96, Kos99]. This sets-as-points approach holds that a fuzzy set is a point in a unit hypercube and a non-fuzzy set is a corner of the hypercube. Normal engineering practice often visualizes binary logical values as the corners of a hypercube, but only fuzzy theory uses the inside of the cube. Fuzzy logic is a natural filling-in of traditional set theory. Any engineer will recognize the 3D representation of all possible combinations three Boolean values: $\{0, 0, 0\}$, $\{0, 0, 1\}$, $\{0, 1, 0\}$, $\{0, 1, 1\}$, $\{1, 0, 0\}$, $\{1, 0, 1\}$, $\{1, 1, 0\}$, $\{1, 1, 1\}$, which correspond to the corners of the unit hypercube. But fuzzy logic also allows any other fractional values inside the hypercube, such as $\{0.5, 0.7, 0.3\}$ corresponding to degrees of set membership.

Fuzzy logic holds that any point inside the unit hypercube is a fuzzy set with Russell’s paradox located at the point of maximum ambiguity in the center of the hypercube.

Fuzzy Entropy

Degrees of fuzziness are referred to as entropy by Kosko. *Fuzzy mutual entropy* measures the *ambiguity of a situation*, information and entropy are inversely related – if you have a maximum-entropy solution, then you have a minimum-information solution, and visa versa, according to Kosko. But minimum-information does not mean that too little information is being used. On the contrary, the principle of maximum entropy ensures that only the relevant information is being used.

This idea of maximizing entropy, according to Kosko, is present throughout the sciences, although it is called by different names. ‘From the quantum level up to astrophysics or anywhere in-between for pattern recognition, you want to use all and only the available information,’ Kosko claims. This emergent model proposes that scientists and engineers estimate the uncertainty structure of a given environment and maximize the entropy relative to the known information, similar to the Lagrange technique in mathematics. The principle of maximum entropy states that any other technique has to be biased, because it has less entropy and thus uses more information than is really available.

Fuzzy theory provides a measure of this entropy factor. It measures ambiguity with operations of union $\cup$, intersection $\cap$ and complement $\bar{}$.

In traditional logic, these three operators are used to define a set of axioms that were proposed by Aristotle to be the immutable laws of (rational) thought, namely, the principle of *non-contradiction* and the principle of the *excluded middle*. The principle of non-contradiction, that nothing can be both A and $\bar{A}$, and the law of the excluded middle, that anything must be either A or $\bar{A}$, amounts to saying that the intersection of a set and its complement is always empty and that the union of a set and its complement always equals the whole *universe of discourse*, respectively. But if we do not know A with certainty, then we do not know $\bar{A}$ with certainty either, else by double negation we would know A with certainty. This produces non-degenerate *overlap* ($A \cap \bar{A}$), which breaks the law of non-contradiction. Equivalently, it also produced non-degenerate *underlap* ($A \cup \bar{A}$) which breaks the law of the excluded middle. In fuzzy logic both these so-called laws are denied. A set and its complement can both be overlap and underlap.

What is worse, there is usually ambiguity in more than one parameter or dimension of a problem. To represent multi-dimensional ambiguity, Kosko shows fuzzy entropy geometrically with a hypercube.

All these relationships are needed in fuzzy logic to express its basic structures for *addition*, *multiplication*, and most important, *implication* $IF \Rightarrow THEN$. They all follow from the subethood relationships between fuzzy sets. The subset relation by itself, corresponds to the implication relation in crisp logic. For instance, $A \Rightarrow B$ is *false only* if the *antecedent* A is *true* and the *consequent* B is *false*. The same holds for subsets, A is a subset of B if there is no element that belongs to A but not to B .

But in fuzzy logic, degrees of subsethood permit some A to be somewhat of a subset of B even though some of its elements are not elements of B . The degree to which A is a subset of B can be measured as the distance from the origin to $(A \cap B)$ divided by the distance from the origin to A .

This structure is derived as a theorem of fuzzy logic, whereas for probability theory equivalent conditional probability theorem has to be assumed, making fuzzy logic a more fundamental.

The *fuzzy mutual entropy* measures how close a fuzzy description of the world is to its own opposite [Kos99]. It has no random analogue in general. The *fuzzy fluid* leads to a type of wave equation. The wave shows how the *extended Shannon entropy potential* $S : [0, 1]^n \rightarrow \mathbb{R}$, defined on the *entire fuzzy cube* $[0, 1]^n$, fluctuates in time. It has the form of a *reaction-diffusion* equation

$$\dot{S} = -c \nabla^2 S, \quad (2.140)$$

where c is the *fuzzy diffusion parameter*. The *fuzzy wave equation* (2.140) implies $\dot{S} > 0$, and thus resembles the entropy increase of the S -theorem of the *Second Law of thermodynamics*.

Similar equations occur in all branches of science and engineering. The Schrödinger wave equation (see [II06a, II06b]) has this form, as well as most models of diffusion. The fuzzy wave equation (2.140) assumes only that information is conserved. The total amount of information is fixed and we do not create or destroy information. Some form of the wave equation would still apply if information were conserved locally or in small regions of system space. The space itself is a fuzzy cube of high dimension. It has as many dimensions as there are objects of interest. The Shannon entropy S changes at each point in this cube and defines a *fuzzy wave*. The obvious result is that the entropy S can only grow in time in the spirit of the second law.

The entropy always grows but its *rate of growth* depends on the system's position in the *fuzzy parameter space*. A deeper result is that entropy changes slowest at the fuzzy cube *midpoint of maximum fuzz*. That is the only point in the cube where the fuzzy description equals its own opposite. The Shannon entropy wave grows faster and faster away from the cube midpoint and near its skin. The skin or surface of the fuzzy cube is the only place where a 0 or 1 appears in the system description. The fuzzy wave equation (2.140) shows that the entropy S changes infinitely fast iff it touches the cube's skin. However, this is impossible in a universe with finite bounds on velocity like the speed of light. So, the result is never a *bit* – it is always a *fit* [Kos99].

Fuzzy Patches for System Modeling

Like ANNs, the fuzzy logic systems are generic *function approximators* [Kos92]. Namely, fuzzy system modeling is performed as a *nonlinear function approximation* using the so-called *fuzzy patches* (see Figure 2.56), which approximate the given function $y = f(x)$, i.e., the *system input-output relation*. The fuzzy patches R_i are given by a set of canonical fuzzy IF-THEN rules:

$$\begin{aligned}
R_1 &: IF \ x \ is \ A_1 \ THEN \ y \ is \ R_1, \\
R_2 &: IF \ x \ is \ A_2 \ THEN \ y \ is \ R_2, \\
&\vdots \\
R_n &: IF \ x \ is \ A_n \ THEN \ y \ is \ R_n.
\end{aligned}$$

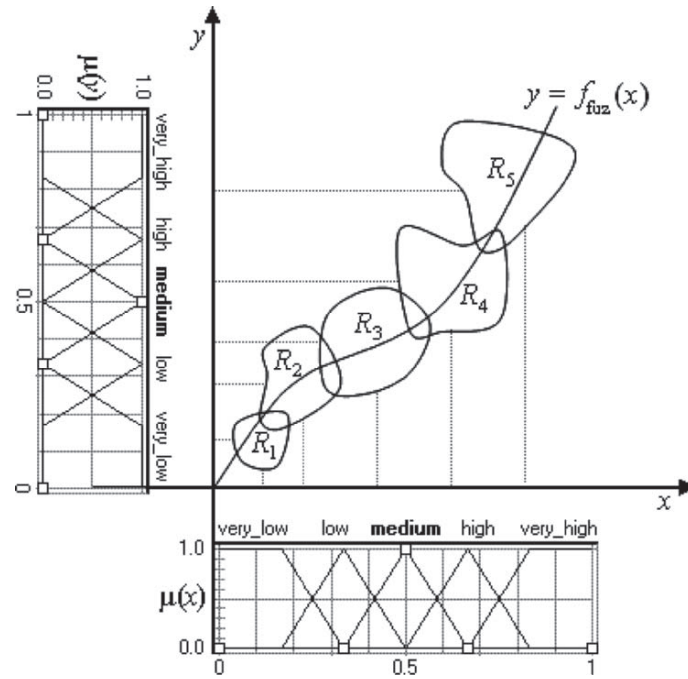


Fig. 2.56. Fuzzy-logic approximation $y = f_{\text{fuz}}(x)$ of an arbitrary function $y = f(x)$ using *fuzzy patches* R_i given by a set of canonical fuzzy IF-THEN rules.

2.6.2 Fuzzy Inference Engine

In the realm of fuzzy logic the above generic nonlinear function approximation is performed by means of fuzzy inference engine. The *fuzzy inference engine* is an *input-output dynamical system* which *maps* a set of input linguistic variables (*IF*-part) into a set of output linguistic variables (*THEN*-part). It has three sequential modules (see Figure 2.57):

1. *Fuzzification*; in this module numerical crisp input variables are fuzzified; this is performed as an overlapping partition of their universes of discourse

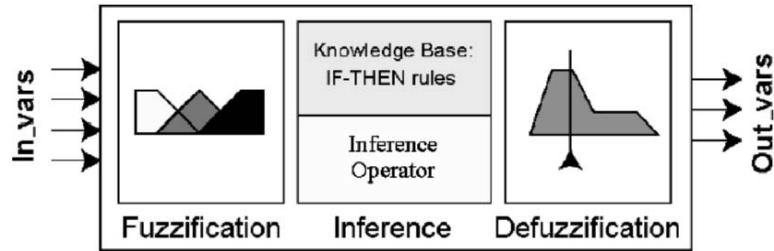


Fig. 2.57. Basic structure of the fuzzy inference engine.

by means of fuzzy membership functions $\mu(x)$ (2.139), which can have various shapes, like triangular–trapezoidal (see Figure 2.58), Gaussian–bell, $\mu(x) = \exp\left[\frac{-(x-m)^2}{2\sigma^2}\right]$ (with mean m and standard deviation σ), sigmoid $\mu(x) = \left[1 + \left(\frac{x-m}{\sigma}\right)^2\right]^{-1}$, or some other shapes.

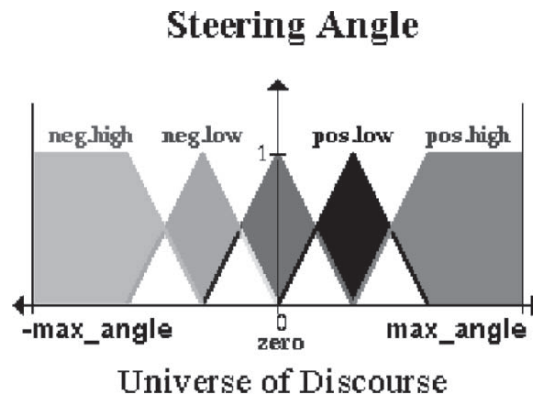


Fig. 2.58. Fuzzification example: set of triangular–trapezoidal membership functions partitioning the universe of discourse for the angle of the hypothetical steering wheel; notice the white overlapping triangles.

B. Kosko and his students have done extensive computer simulations looking for the best shape of fuzzy sets to model a known test system as closely as possible. They let fuzzy sets of all shapes and sizes compete against each other. They also let neural systems tune the fuzzy–set curves to improve how well they model the test system. The main conclusion from these experiments is that ‘triangles never do well’ in such contests. Suppose we want an adaptive fuzzy system $F : \mathbb{R}^n \rightarrow \mathbb{R}$ to approximate a test function (or, approximand) $f : \mathbb{R}^n \rightarrow \mathbb{R}$ as closely as possible in the sense of minimizing the mean–squared error between them, $\left(\|f - F\|^2\right)$. Then the i th scalar ‘sinc’ function (as commonly used in signal processing),

$$\mu_i(x) = \frac{\sin\left(\frac{x-m_i}{d_i}\right)}{\frac{x-m_i}{d_i}}, \quad (i = 1, \dots, n), \quad (2.141)$$

with center m_i and dispersion (width) $d_i = \sigma_i^2 > 0$, often gives the best performance for *IF*-part mean-squared function approximation, even though this generalized function can take on negative values (see [Kos99]).

2. *Inference*; this module has two submodules:

- (i) The expert-knowledge base consisting of a set of *IF* – *THEN* rules relating input and output variables, and
- (ii) The inference method, or implication operator, that actually combines the rules to give the fuzzy output; the most common is *Mamdani Min-Max inference*, in which the membership functions for input variables are first combined inside the *IF* – *THEN* rules using *AND* ($\cap$, or *Min*) operator, and then the output fuzzy sets from different *IF* – *THEN* rules are combined using *OR* ($\cup$, or *Max*) operator to get the common fuzzy output (see Figure 2.59).

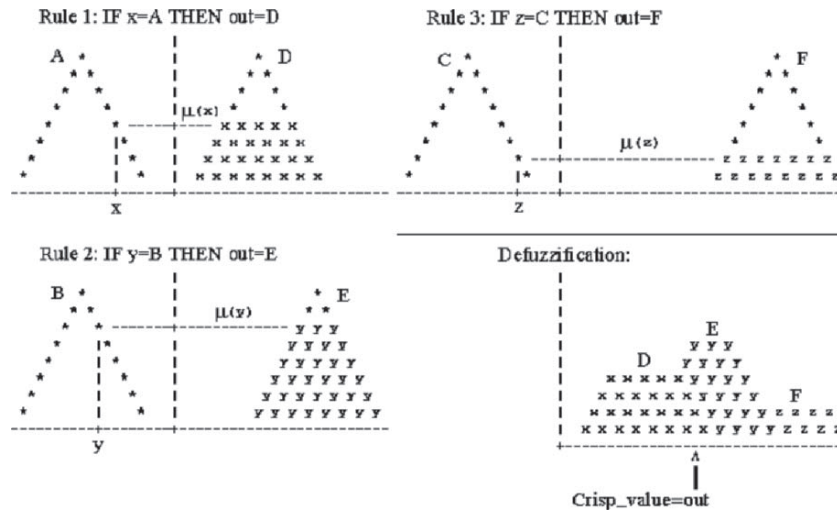


Fig. 2.59. Mamdani's Min-Max inference method and Center of Gravity defuzzification.

3. *Defuzzification*; in this module fuzzy outputs from the inference module are converted to numerical crisp values; this is achieved by one of the several defuzzification algorithms; the most common is the Center of Gravity method, in which the crisp output value is calculated as the abscissa under the center of gravity of the output fuzzy set (see Figure 2.59).

In more complex technical applications of general function approximation (like in complex control systems, signal and image processing, etc.), two

optional blocks are usually added to the fuzzy inference engine [Kos92, Kos96, Lee90]:

(0) *Preprocessor*, preceding the fuzzification module, performing various kinds of normalization, scaling, filtering, averaging, differentiation or integration of input data; and

(4) *Postprocessor*, succeeding the defuzzification module, performing the analog operations on output data.

Common fuzzy systems have a simple feedforward mathematical structure, the so-called *Standard Additive Model* (SAM), which aids the spread of applications. Almost all applied fuzzy systems use some form of SAM, and some SAMs in turn resemble the ANN models (see [Kos99]).

In particular, an *additive fuzzy system* $F : \mathbb{R}^n \rightarrow \mathbb{R}^p$ stores m rules of the patch form $A_i \times B_i \subset \mathbb{R}^n \times \mathbb{R}^p$, or of the word form ‘**If** $X = A_i$ **Then** $Y = B_i$ ’ and adds the ‘fired’ Then-parts $B'_i(x)$ to give the output set $B(x)$, calculated as

$$B(x) = w_i B'_i(x) = w_i \mu_i(x) B_i(x), \quad (i = 1, \dots, n), \quad (2.142)$$

for a scalar rule weight $w_i > 0$. The factored form $B'_i(x) = \mu_i(x) B_i(x)$ makes the additive system (2.142) a SAM system. The fuzzy system F computes its output $F(x)$ by taking the centroid of the output set $B(x)$: $F(x) = \text{Centroid}(B(x))$. The *SAM theorem* then gives the centroid as a simple ratio,

$$F(x) = p_i(x) c_i, \quad (i = 1, \dots, n),$$

where the convex coefficients or discrete probability weights $p_i(x)$ depend on the input x through the ratios

$$p_i(x) = \frac{w_i \mu_i(x) V_i}{w_k \mu_k(x) V_k}, \quad (i = 1, \dots, n). \quad (2.143)$$

V_i is the finite positive volume (or area if $p = 1$ in the codomain space $\mathbb{R}^p$) [Kos99],

$$V_i = \int_{\mathbb{R}^p} b_i(y_1, \dots, y_p) dy_1 \dots dy_p > 0,$$

and c_i is the centroid of the Then-part set $B_i(x)$,

$$c_i = \frac{\int_{\mathbb{R}^p} y b_i(y_1, \dots, y_p) dy_1 \dots dy_p}{\int_{\mathbb{R}^p} b_i(y_1, \dots, y_p) dy_1 \dots dy_p}.$$

2.6.3 Fuzzy Logic Control

The most common and straightforward applications of fuzzy logic are in the domain of nonlinear control [Kos92, Kos96, Lee90, DSS96]. Fuzzy control is a nonlinear control method based on fuzzy logic. Just as fuzzy logic can be described simply as computing with words rather than numbers, fuzzy control

can be described simply as control with sentences rather than differential equations.

A fuzzy controller is based on the fuzzy inference engine, which acts either in the feedforward or in the feedback path, or as a supervisor for the conventional PID controller.

A fuzzy controller can work either directly with fuzzified dynamical variables, like direction, angle, speed, or with their fuzzified errors and rates of change of errors. In the second case we have rules of the form:

1. IF error is *Neg* AND change in error is *Neg* THEN output is *NB*.
2. IF error is *Neg* AND change in error is *Zero* THEN output is *NM*.

The collection of rules is called a rule base. The rules are in *IF – THEN* format, and formally the *IF*–side is called the condition and the *THEN*–side is called the conclusion (more often, perhaps, the pair is called antecedent – consequent). The input value *Neg* is a linguistic term short for the word Negative, the output value *NB* stands for *Negative_Big* and *NM* for *Negative_Medium*. The computer is able to execute the rules and compute a control signal depending on the measured inputs error and change in error.

The rule–base can be also presented in a convenient form of one or several rule matrices, the so–called *FAM*–matrices, where *FAM* is a shortcut for Kosko’s *fuzzy associative memory* [Kos92, Kos96]. For example, a 9×9 graded FAM matrix can be defined in a symmetrical weighted form:

$$\text{FAM} = \begin{pmatrix} 0.6\text{S4} & 0.6\text{S4} & 0.7\text{S3} & \dots & \text{CE} \\ 0.6\text{S4} & 0.7\text{S3} & 0.7\text{S3} & \dots & 0.9\text{B1} \\ 0.7\text{S3} & 0.7\text{S3} & 0.8\text{S2} & \dots & 0.9\text{B1} \\ \dots & \dots & \dots & \dots & 0.6\text{B4} \\ \text{CE} & 0.9\text{B1} & 0.9\text{B1} & \dots & 0.6\text{B4} \end{pmatrix},$$

in which the vector of nine linguistic variables L^9 partitioning the *universes of discourse* of all three variables (with trapezoidal or Gaussian bell–shaped *membership functions*) has the form

$$L^9 = \{\text{S4}, \text{S3}, \text{S2}, \text{S1}, \text{CE}, \text{B1}, \text{B2}, \text{B3}, \text{B4}\}^T,$$

to be interpreted as: ‘small 4’, ... , ‘small 1’, ‘center’, ‘big 1’, ... , ‘big 4’. For example, the left upper entry (1,1) of the FAM matrix means: IF red is S4 and blue is S4, THEN result is 0.6S4; or, entry (3,7) means: IF red is S2 and blue is B2, THEN result is center, etc.

Here we give three design examples for fuzzy controllers, the first one in detail, and the other two briefly.

Mamdani Fuzzy Controller for Balancing an Inverted Pendulum

The problem is to balance θ a pole of mass m and inertia moment I on a mobile platform of mass M that can be forced by F to move only (left/right)

along x -axis (see Figure 2.60). This is quite an involved problem for conventional PID controller, based on differential equations of the pole and platform motion. Instead, we will apply fuzzy linguistic technique called *Mamdani inference* (see previous subsection).

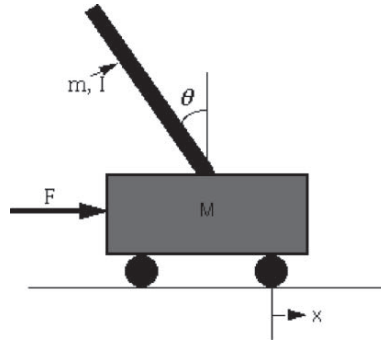


Fig. 2.60. Problem of balancing an inverted pendulum.

Firstly, as a *fuzzification* part, we have to define (subjectively) what high speed, low speed etc. of the platform M is. This is done by specifying the membership functions for the fuzzy set partitions of the *platform speed* universe of discourse, using the following linguistic variables: (i) negative high (NH), (ii) negative low (NL), (iii) zero (ZE), (iv) positive low (PL), and (v) positive high (PH) (see Figure 2.61).³⁵

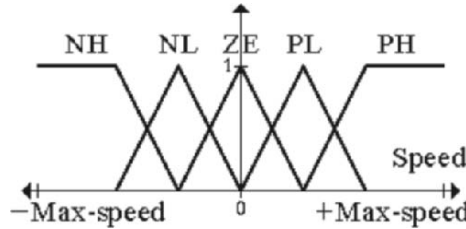


Fig. 2.61. Fuzzy membership functions for speed of the platform.

Also, we need to do the same for the angle θ between the platform and the pendulum and the angular velocity $\dot{\theta}$ of this angle (see Figure 2.62).

Secondly, as an *inference* part, we give several *fuzzy IF-THEN rules* that will tell us what to do in certain situations. Consider for example that the

³⁵ For simplicity, we assume that in the beginning the pole is in a nearly upright position so that an angle θ greater than, 45 degrees in any direction can never occur.

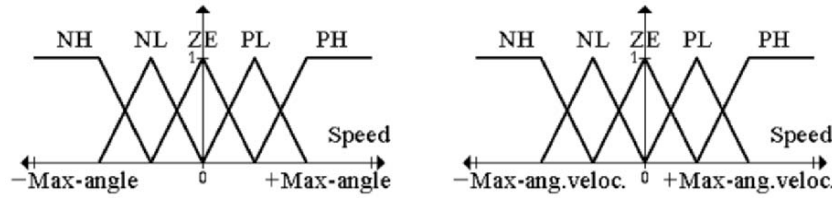


Fig. 2.62. Fuzzy membership functions for speed of the platform.

pole is in the upright position (angle θ is zero) and it does not move (angular velocity $\dot{\theta}$ is zero). Obviously this is the desired situation, and therefore we don't have to do anything (speed is zero). Let us consider also another case: the pole is in upright position as before but is in motion at *low* velocity in *positive* direction. Naturally we would have to compensate the pole's movement by moving the platform in the same direction at *low* speed.

So far we've made up two rules that can be put into a more formalized form like this:

IF angle is zero AND angular velocity is zero THEN speed shall be zero.

IF angle is zero AND angular velocity is positive low THEN speed shall be positive low.

We can summarize all applicable rules in the following FAM table (see previous subsection):

Speed		Angle				
		NH	NL	ZE	PL	PH
V	NH			NH		
e	NL			NL	ZE	
l	ZE	NH	NL	ZE	PL	PH
o	PL		ZE	PL		
c	PH			PH		

Now, we are going to define two explicit values for angle and angular velocity to calculate with. Consider the situation given in Figure 2.63, and let us apply the following rule:

IF angle is zero AND angular velocity is zero THEN speed is zero

– to the values that we have previously selected (see Figure 2.64)

Only four rules yield a result (*rules fire*, see Figure 2.65), and we overlap them into one single result (see Figure 2.66).

Fan: the Temperature Control System

In this simple example, the input linguistic variable is:

$$temperature\_error = desired\_temperature - current\_temperature.$$

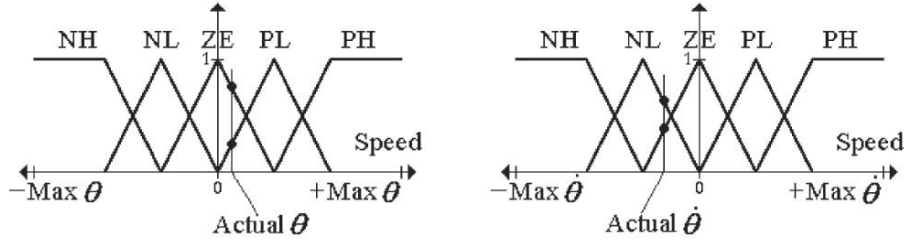


Fig. 2.63. Actual values for angle θ and angular velocity $\dot{\theta}$.

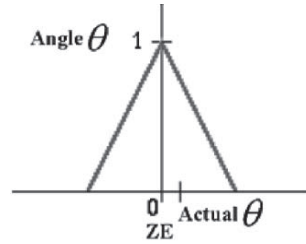


Fig. 2.64. Here is the linguistic variable *angle* θ where we zoom-in on the fuzzy set zero (ZE) and the actual angle.

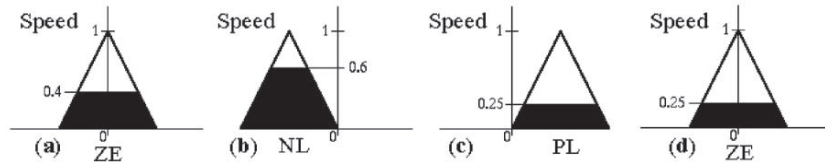


Fig. 2.65. Four fuzzy rules firing: (a) the result yielded by the rule: IF angle is zero AND angular velocity is zero THEN speed is zero; (b) the result yielded by the rule: IF angle is zero AND angular velocity is negative low THEN speed is negative low; (c) the result yielded by the rule: IF angle is positive low AND angular velocity is zero THEN speed is positive low; (d) the result yielded by the rule: IF angle is positive low AND angular velocity is negative low THEN speed is zero.

The two output linguistic variables are:

hot_fan_speed, and *cool_fan_speed*. The universes of discourse, consisting of membership functions, i.e., overlapping triangular–trapezoidal shaped intervals, for all three variables are:

invar: temperature_error = {*Negative_Big*, *Negative_Medium*, *Negative_Small*, *Zero*, *Positive_Small*, *Positive_Medium*, *Positive_Big*}, with the range $[-110, 110]$ degrees;

outvars: hot_fan_speed and *cool_fan_speed* = {*zero*, *low*, *medium*, *high*, *very_high*}, with the range $[0, 100]$ rounds-per-meter.

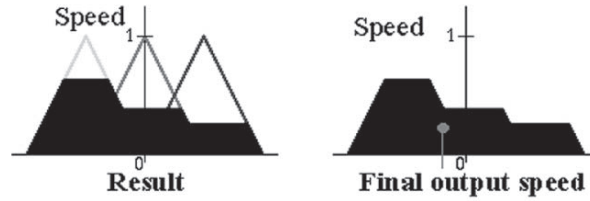


Fig. 2.66. Left: Overlapping single-rule results to yield the overall result. Right: The result of the fuzzy controller so far is a fuzzy set (of speed), so we have to choose one representative value as the final output; there are several heuristic *defuzzification* methods, one of them is to take the center of gravity of the fuzzy set. This is called *Mamdani fuzzy controller*.

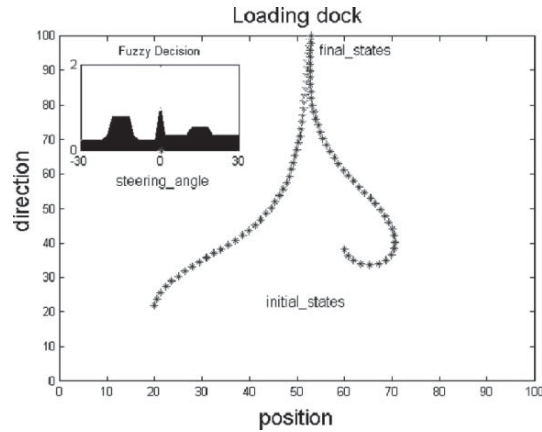


Fig. 2.67. Truck backer-upper steering control system.

Truck Backer-Upper Steering Control System

In this example there are two input linguistic variables: position and direction of the truck, and one output linguistic variable: steering angle (see Figure 2.67). The universes of discourse, partitioned by overlapping triangular-trapezoidal shaped intervals, are defined as:

invars: $position = \{NL, NS, ZR, PS, PL\}$, and
 $direction = \{NL, NM, NS, ZR, PS, PM, PL\}$, where *NL* denotes Negative_Large, *NM* is Negative_Medium, *NS* is Negative_Small, etc.

outvar: $steering\_angle = \{NL, NM, NS, ZR, PS, PM, PL\}$.

The rule-base is given as:

IF direction is NL, AND position is NL, THEN steering angle is NL;
 IF direction is NL, AND position is NS, THEN steering angle is NL;
 IF direction is NL, AND position is ZE, THEN steering angle is PL;
 IF direction is NL, AND position is PS, THEN steering angle is PL;

IF direction is NL, AND position is PL, THEN steering angle is PL;
 IF direction is NM, AND position is NL, THEN steering angle is ZE;

 IF direction is PL AND position is PL, THEN steering angle is PL.

The so-called *control surface* for the truck backer-upper steering control system is depicted in Figure 2.68.

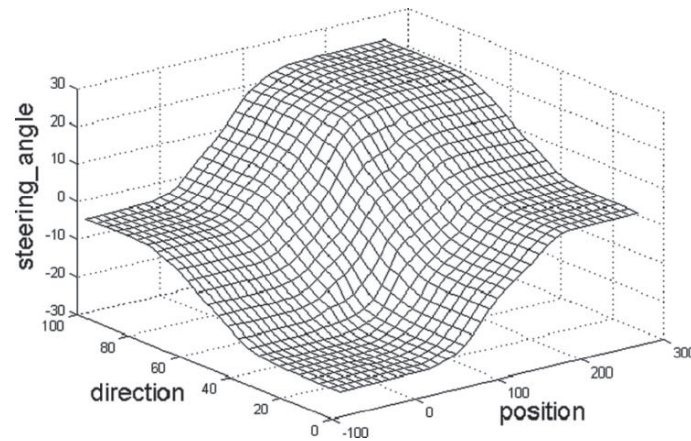


Fig. 2.68. Control surface for the truck backer-upper steering control system.

To distinguish between more and less important rules in the knowledge base, we can put weights on them. Such weighted knowledge base can be then trained by means of artificial neural networks. In this way we get *hybrid neuro-fuzzy trainable expert systems*.

Another way of the hybrid neuro-fuzzy design is the fuzzy inference engine such that each module is performed by a layer of hidden artificial neurons, and ANN-learning capability is provided to enhance the system knowledge (see Figure 2.69).

Again, the fuzzy control of the BP learning (2.116–2.117) can be implemented as a set of heuristics in the form of fuzzy *IF – THEN* rules, for the purpose of achieving a faster rate of convergence. The heuristics are driven by the behavior of the instantaneous sum of squared errors.

Finally, most *feedback fuzzy systems* are either discrete or continuous generalized SAMs [Kos99], given respectively by

$$x(k+1) = p_i(x(k))B_i(x(k)), \quad \text{or} \quad \dot{x}(t) = p_i(x(t))B_i(x(t)),$$

with coefficients p_i given by (2.143) above.

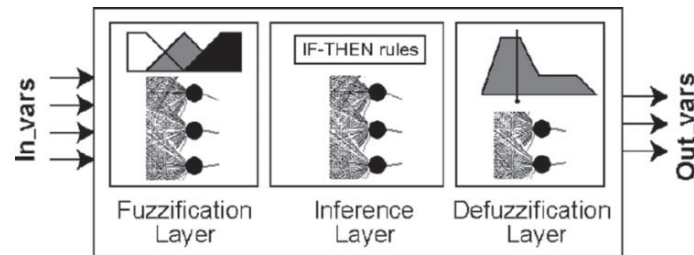


Fig. 2.69. Neuro-fuzzy inference engine.

General Characteristics of Fuzzy Control

As demonstrated above, fuzzy logic offers several unique features that make it a particularly good choice for many control problems, among them [Lee90, DSS96]:

1. It is inherently robust since it does not require precise, noise-free inputs and can be programmed to fail safely if a feedback sensor quits or is destroyed. The output control is a smooth control function despite a wide range of input variations.
2. Since the fuzzy logic controller processes user-defined rules governing the target control system, it can be modified and tweaked easily to improve or drastically alter system performance. New sensors can easily be incorporated into the system simply by generating appropriate governing rules.
3. Fuzzy logic is not limited to a few feedback inputs and one or two control outputs, nor is it necessary to measure or compute rate-of-change parameters in order for it to be implemented. Any sensor data that provides some indication of a systems actions and reactions is sufficient. This allows the sensors to be inexpensive and imprecise thus keeping the overall system cost and complexity low.
4. Because of the rule-based operation, any reasonable number of inputs can be processed (1–8 or more) and numerous outputs (1–4 or more) generated, although defining the rule-base quickly becomes complex if too many inputs and outputs are chosen for a single implementation since rules defining their interrelations must also be defined. It would be better to break the control system into smaller chunks and use several smaller fuzzy logic controllers distributed on the system, each with more limited responsibilities.
5. Fuzzy logic can control nonlinear systems that would be difficult or impossible to model mathematically. This opens doors for control systems that would normally be deemed unfeasible for automation.

A *fuzzy logic controller* is usually designed using the following steps:

1. Define the control objectives and criteria: What am I trying to control? What do I have to do to control the system? What kind of response do I need? What are the possible (probable) system failure modes?
2. Determine the input and output relationships and choose a minimum number of variables for input to the fuzzy logic engine (typically error and rate-of-change of error).
3. Using the rule-based structure of fuzzy logic, break the control problem down into a series of *IF X AND Y THEN Z* rules that define the desired system output response for given system input conditions. The number and complexity of rules depends on the number of input parameters that are to be processed and the number fuzzy variables associated with each parameter. If possible, use at least one variable and its time derivative. Although it is possible to use a single, instantaneous error parameter without knowing its rate of change, this cripples the systems ability to minimize overshoot for a step inputs.
4. Create fuzzy logic membership functions that define the meaning (values) of Input/Output terms used in the rules.
5. Test the system, evaluate the results, tune the rules and membership functions, and re-test until satisfactory results are obtained.

Therefore, fuzzy logic does not require precise inputs, is inherently robust, and can process any reasonable number of inputs but system complexity increases rapidly with more inputs and outputs. Distributed processors would probably be easier to implement. Simple, plain-language rules of the form *IF X AND Y THEN Z* are used to describe the desired system response in terms of linguistic variables rather than mathematical formulas. The number of these is dependent on the number of inputs, outputs, and the designers control response goals. Obviously, for very complex systems, the rule-base can be enormous and this is actually the only drawback in applying fuzzy logic.

Evolving Fuzzy-Connectionist Systems

Recently, [Kas02] introduced a new type of fuzzy inference systems, denoted as dynamic evolving neuro-fuzzy inference system (DENFIS), for adaptive online and off-line learning, and their application for dynamic time series prediction. *DENFIS system* evolves through incremental, hybrid (supervised/unsupervised), learning, and accommodates new input data, including new features, new classes, etc., through local element tuning. New fuzzy rules are created and updated during the operation of the system. At each time moment, the output of DENFIS is calculated through a fuzzy inference system based on *m*-most activated fuzzy rules which are dynamically chosen from a fuzzy rule set. Two approaches are proposed: (i) dynamic creation of a first-order Takagi-Sugeno-type (see, e.g., [TW01]) fuzzy rule set for a DENFIS online model; and (ii) creation of a first-order Takagi-Sugeno-type fuzzy rule set, or an expanded high-order one, for a DENFIS offline model. A set of

fuzzy rules can be inserted into DENFIS before or during its learning process. Fuzzy rules can also be extracted during or after the learning process. An evolving clustering method (ECM), which is employed in both online and off-line DENFIS models, is also introduced. It was demonstrated that DENFIS could effectively learn complex temporal sequences in an adaptive way and outperform some well-known, existing models.

2.7 Synergetics Approach

In this section we give a brief review on synergetics approach to brain and cognition. Formally, here we are dealing with a *synergetics evolution* 2-functor $\mathcal{E}$, given by

$$\begin{array}{ccc}
 \begin{array}{ccc}
 A & \xrightarrow{f} & B \\
 \downarrow h & \text{CURRENT} & \downarrow g \\
 & \text{SYNERGY} & \\
 & \text{STATE} & \\
 C & \xrightarrow{k} & D
 \end{array}
 & \xrightarrow{\mathcal{E}} &
 \begin{array}{ccc}
 \mathcal{E}(A) & \xrightarrow{\mathcal{E}(f)} & \mathcal{E}(B) \\
 \downarrow \mathcal{E}(h) & \text{DESIRED} & \downarrow \mathcal{E}(g) \\
 & \text{SYNERGY} & \\
 & \text{STATE} & \\
 \mathcal{E}(C) & \xrightarrow{\mathcal{E}(k)} & \mathcal{E}(D)
 \end{array}
 \end{array}$$

Here $\mathcal{E}$ represents an association/projection functor from the *source* 2-category of the current synergetics state, defined as a commutative square of small synergetics categories $A, B, C, D, \dots$ of *current synergetics components* and their causal interrelations $f, g, h, k, \dots$, onto the *target* 2-category of the desired synergetics state, defined as a commutative square of small synergetics categories $\mathcal{E}(A), \mathcal{E}(B), \mathcal{E}(C), \mathcal{E}(D), \dots$ of *evolved synergetics components* and their causal interrelations $\mathcal{E}(f), \mathcal{E}(g), \mathcal{E}(h), \mathcal{E}(k)$. As in the previous section, each causal arrow in above diagram, e.g., $f : A \rightarrow B$, stands for a *generic synergetics dynamorphism*.

Recall that *synergetics* is an interdisciplinary field of research that was founded by H. Haken in 1969 (see [Hak83, Hak93, Hak96, Hak00]). Synergetics deals with complex systems that are composed of many individual parts (components, elements) that interact with each other and are able to produce spatial, temporal or functional structures by self-organization. In particular, synergetics searches for general principles governing self-organization irrespective of the nature of the individual parts of the systems that may belong to a variety of disciplines such as physics (lasers, fluids, plasmas), meteorology, chemistry (pattern formation by chemical reactions, including flames), biology (morphogenesis, evolution theory) movement science, brain activities, computer sciences (synergetic computer), sociology (e.g., city growth) psychology and psychiatry (including Gestalt psychology).

The aim of synergetics has been to describe processes of *spontaneous self-organization and cooperation* in complex systems built from many subsystems

which themselves can be complicated nonlinear objects (like many individual neuro-muscular components of the human motion system, having their own excitation and contraction dynamics, embedded in a synergistic way to produce coordinated human movement). General properties of the subsystems are their own nonlinear/chaotic dynamics as well as mutual nonlinear/chaotic interactions. Furthermore, the systems of synergetics are *open*. The influence from outside is measured by a certain set of *control parameters* $\{\sigma\}$ (like amplitudes, frequencies and time characteristics of neuro-muscular driving forces). Processes of self-organization in synergetics, (like musculo-skeletal coordination in human motion dynamics) are observed as temporal macroscopic patterns. They are described by a small set of *order parameters* $\{o\}$, similar to those in Landau's *phase-transition theory* (named after *Nobel Laureate Lev D. Landau*) of physical systems in *thermal equilibrium* [Hak83].

Now, recall that the *measure for the degree of disorder* in any isolated, or conservative, system (such a system that does not interact with its surrounding, i.e., does neither dissipate nor gain energy) is *entropy*. The *second law of thermodynamics*³⁶ states that in every conservative irreversible system the entropy ever increases to its maximal value, i.e., to the total disorder of the system (or remains constant for a reversible system).

Example of such a system is conservative Hamiltonian dynamics of human skeleton in the phase-space Γ defined by all joint angles q^i and momenta p_i ³⁷, defined by ordinary (conservative) Hamilton's equations

$$\dot{q}^i = \partial_{p_i} H, \quad \dot{p}_i = -\partial_{q^i} H. \quad (2.144)$$

The basic fact of the conservative Hamiltonian system is that its phase-flow, the time evolution of equations (2.144), preserves the phase-space volume (the so-called *Liouville measure*), as proposed by the *Liouville theorem*. This might look fine at first sight, however, the preservation of phase-space volume causes *structural instability* of the conservative Hamiltonian system, i.e., the phase-space spreading effect, by which small phase regions R_t will tend to get distorted from the initial one R_0 during the system evolution. The problem is much more serious in higher dimensions than in lower dimensions, since there are so many 'directions' in which the region can locally spread. Here we see the work of the second law of thermodynamics on an irreversible process: the increase of entropy towards the total disorder/chaos [Pen89]. In this way, the conservative Hamiltonian systems of the form (2.144) cover the wide range of dynamics, from completely integrable, to completely ergodic. Biodynamics of human-like movement is probably somewhere in the middle of this range, the more DOF included in the model, the closer to the ergodic case. One can easily imagine that the conservative skeleton-like system with 300 DOF, which means 600-D system of the form (2.144), which is full of trigonometry

³⁶ This is the only physical law that implies the arrow of time.

³⁷ If we neglect joints dissipation and muscular driving forces, we are dealing with pure skeleton conservative dynamics.

(coming from its noncommutative rotational matrices), is probably closer to the ergodic than to the completely integrable case.

On the other hand, when we manipulate a system from the outside, by the use of certain *control parameters* $\{\sigma\}$, we can change its *degree of order* (see [Hak83, Hak93]). Consider for example *water vapor*. At elevated temperature its molecules move freely without mutual correlation. When temperature is lowered, a liquid drop is formed, the molecules now keep a mean distance between each other. Their motion is thus highly correlated. Finally, at still lower temperature, at the freezing point, water is transformed into ice crystals. The transitions between the different aggregate states, also called phases, are quite abrupt. Though the same kind of molecules are involved all the time, the macroscopic features of the three phases differ drastically.

Similar type of ordering, but not related to the thermal equilibrium conditions, occurs in *lasers*, mathematically given by Lorenz-like attractor equations. Lasers are certain types of lamps which are capable of emitting coherent light. A typical laser consists of a crystal rod filled with gas, with the following features important from the synergetics point of view: when the atoms the laser material consists of are excited or ‘pumped’ from the outside, they emit light waves. So, the pump power, or pump rate represents the control parameter σ . At low pump power, the waves are entirely uncorrelated as in a usual lamp. Could we hear light, it would sound like noise to us [Hak83].

When we increase the pump rate to a critical value σ_c , the noise disappears and is replaced by a pure tone. This means that the atoms emit a pure sinusoidal light wave which in turn means that the individual atoms act in a perfectly correlated way – they become self-organized. When the pump rate is increased beyond a second critical value, the laser may periodically emit very intense and short pulses. In this way the following *instability sequence* occurs [Hak83]:

noise $\mapsto$ {coherent oscillation at frequency ω_1 } $\mapsto$
periodic pulses at frequency ω_2 which modulate oscillation at frequency ω_1
i.e., no oscillation $\mapsto$ first frequency $\mapsto$ second frequency.

Under different conditions the light emission may become *chaotic* or even *turbulent*. The frequency spectrum becomes broadened.

The laser played a crucial role in the development of synergetics for various reasons [Hak83]. In particular, it allowed detailed theoretical and experimental study of the phenomena occurring within the transition region: *lamp* $\leftrightarrow$ *laser*, where a surprising and far-reaching analogy with phase transitions of systems in thermal equilibrium was discovered. This analogy includes all basic *phase-transition effects*: a *symmetry breaking instability*, *critical slowing down* and *hysteresis effect*.

2.7.1 Phase Transitions

Besides water vapor, a typical example is a *ferromagnet* [Hak83]. When a ferromagnet is heated, it suddenly loses its magnetization. When temperature is lowered, the magnet suddenly regains its magnetization. What happens on a microscopic, atomic level, is this: We may visualize the magnet as being composed of many, elementary (atomic) magnets (called spins). At elevated temperature, the elementary magnets point in random directions. Their magnetic moments, when added up, cancel each other and no macroscopic magnetization results. Below a critical value of temperature T_c , the elementary magnets are lined up, giving rise to a macroscopic magnetization. Thus the *order on the microscopic level* is a cause of a *new feature* of the material *on the macroscopic level*. The change of one phase to the other one is called *phase transition*.

A thermodynamical description of ferromagnet is based on analysis of its *free energy potential* (in thermal equilibrium conditions). The free energy $\mathcal{F}$, depends on the *control parameter* $\sigma = T$, the temperature. We seek the minimum of the potential $\mathcal{F}$ for a fixed value of magnetization o , which is called *order parameter* in Landau's theory of phase transitions.

This phenomenon is called a *phase transition of second order* because the second derivative (specific heat) of the free energy potential $\mathcal{F}$ is discontinuous. On the other hand, the entropy S (the first derivative of $\mathcal{F}$) itself is continuous so that this transition is also referred to as a *continuous phase transition*.

In statistical physics one also investigates the temporal change of the order parameter – magnetization o . Usually, in a more or less phenomenological manner, one assumes that o obeys an equation of the form

$$\dot{o} = -\frac{\partial \mathcal{F}}{\partial o} = -\sigma o - \beta o^3. \quad (2.145)$$

For $\sigma \rightarrow 0$ we observe a phenomenon called *critical slowing down*, because the ‘particle’ with coordinate o falls down the slope of the ‘potential well’ more and more slowly. Simple relation (2.145) is called *order parameter equation*.

We now turn to the case where the free energy potential has the form

$$\mathcal{F}(o, T) = \frac{\sigma}{2} o^2 + \frac{\gamma}{3} o^3 + \frac{\beta}{4} o^4, \quad (2.146)$$

(β and γ – positive but σ may change its sign according to $\sigma = a(T - T_c)$, ($a > 0$)). When we change the control parameter – temperature T , i.e., the parameter σ , we pass through a sequence of deformations of the potential curve.

When lowering temperature, the local minimum first remains at $o_0 = 0$. When lowering temperature, the ‘particle’ may fall down from o_0 to the new (global) minimum of $\mathcal{F}$ at o_1 . The entropies of the two states, o_0 and o_1 , differ. This phenomenon is called a *phase transition of first order* because the first derivative of the potential $\mathcal{F}$ with respect to the control parameter T

is discontinuous. Since the entropy S is discontinuous this transition is also referred to as a *discontinuous phase transition*. When we now increase the temperature, is apparent that the system stays at o_1 longer than it had been before when lowering the control parameter. This represents *hysteresis effect*.

In the case of the potential (2.146) the order parameter equation gets the form

$$\dot{o} = -\sigma o - \gamma o^2 - \beta o^3.$$

Similar *disorder* $\Rightarrow$ *order* transitions occur also in various non-equilibrium systems of physics, chemistry, biology, psychology, sociology, as well as in human motion dynamics. The analogy is subsumed in Table 1.

Table 1. Phase transition analogy

System in thermal equilibrium	Non-equilibrium system
Free energy potential $\mathcal{F}$	Generalized potential V
Order parameters o_i	Order parameters o_i
$\dot{o}_i = -\frac{\partial \mathcal{F}}{\partial o_i}$	$\dot{o}_i = -\frac{\partial V}{\partial o_i}$
Temperature T	Control input u
Entropy S	System output y
Specific Heat c	System efficiency e

In the case of human motion dynamics, natural control inputs u_i are muscular torques F_i , natural system outputs y_i are joint coordinates q^i and momenta p_i , while the system efficiencies e_i represent the changes of coordinates and momenta with changes of corresponding muscular torques for the i th joint,

$$e_i^q = \frac{\partial q^i}{\partial F_i}, \quad e_i^p = \frac{\partial p_i}{\partial F_i}.$$

Order parameters o_i represent certain important qualities of the human motion system, depending on muscular torques as control inputs, similar to *magnetization*, and usually defined by equations similar to (2.145) or

$$\dot{o}_i = -\sigma o - \gamma o^2 - \beta o^3,$$

with nonnegative parameters σ, β, γ , and corresponding to the second and first order phase transitions, respectively. The choice of actual order parameters is a matter of *expert knowledge* and *purpose of macroscopic system modeling* [Hak83].

2.7.2 Hamiltonian Biodynamics

Now, we turn to biodynamics and muscular synergy. To describe the biodynamics of human-like movement, namely our *covariant force law*, $F_i = mg_{ij}a^j$ (see [II05, II06a, II06b]), we can also start from generalized Hamiltonian vector-field X_H describing the behavior of the *human-like locomotor system*

$$\dot{q}^i = \frac{\partial H}{\partial p_i} + \frac{\partial R}{\partial p_i}, \quad (2.147)$$

$$\dot{p}_i = F_i - \frac{\partial H}{\partial q^i} + \frac{\partial R}{\partial q^i}, \quad (2.148)$$

where the vector-field X_H is generating time evolution, or *phase-flow*, of $2n$ *system variables*: n generalized coordinates (joint angles q^i) and n generalized momenta (joint angular momenta p_i), $H = H(q, p)$ represents the system's conservative energy: kinetic energy + various mechano-chemical potentials, $R = R(q, p)$ denotes the nonlinear dissipation of energy, and $F_i = F_i(t, q, p, \sigma)$ are external control forces (biochemical energy inputs). The *system parameters* include inertia tensor with mass distribution of all body segments, stiffness and damping tensors for all joints (labeled by index i , which is, for geometric reasons, written as a subscript on angle variables, and as a superscript on momentum variables), as well as amplitudes, frequencies and time characteristics of all active muscular forces (supposed to be acting in all the joints; if some of the joints are inactive, we have the affine Hamiltonian control system, see chapter 6).

The equation (2.147) is called the *velocity equation*, representing the *flow* of the system (analogous to current in electrodynamics), while the equation (2.148) is a Newton-like *force equation*, representing the *effort* of the system (analogous to voltage). Together, these two functions represent Hamiltonian formulation of the *biomechanical force-velocity relation* of A.V. Hill [Hil38]. From engineering perspective, their (inner) product, *flow · effort*, represents the total system's *power*, equal to the time-rate-of-change of the total system's energy (included in H, R and F_i functions). And energy itself is transformed into the *work* done by the system.

Now, the reasonably accurate musculo-skeletal biodynamics would include say a hundred DOF, which means a hundred of joint angles and a hundred of joint momenta, which further means a hundred of coupled equations of the form of (2.147–2.148). And the *full coupling* means that each angle (and momentum) includes the information of all the other angles (and momenta), the *chain coupling* means that each angle (and momentum) includes the information of all the previous (i.e., children) angles (and momenta), the *nearest neighbor coupling* includes the information of the nearest neighbors, etc.

No matter which coupling we use for modeling the dynamics of human motion, one thing is certain: the *coupling is nonlinear*. And we obviously have to fight chaos within several hundreds of variables.

Wouldn't it be better if we could somehow be able to obtain a synthetic information about the whole musculo-skeletal dynamics, synthesizing the hundreds of equations of motion of type (2.147–2.148) into a small number of equations describing the time evolution of the so-called *order parameters*? If we could do something similar to principal component analysis in multivariate statistics and neural networks, to get something like 'nonlinear factor dynamics'?

Starting from the basic system (2.147–2.148), on the lowest, *microscopic level of human movement organization*, the *order parameter equations of macroscopic synergetics* can be (at least theoretically), either exactly derived along the lines of *mezosopic synergetics*, or phenomenologically stated by the use of the certain biophysical analogies and nonlinear identification and control techniques (a highly complex nonlinear system like human locomotor apparatus could be neither identified nor controlled by means of standard linear engineering techniques).

2.7.3 Kick Dynamics and Dissipation–Fluctuation Theorem

Deterministic Delayed Kicks

Following [Hak02], we consider the mechanical example of a soccer ball that is kicked by a soccer player and rolls over grass, whereby its motion will be slowed down. We start with the Newton's (second) law of motion, $m\dot{v} = \text{force}$, and in order to get rid of superfluous constants, we put temporarily $m = 1$. The *force* on the r.h.s. consists of the damping force $-\gamma v(t)$ of the grass (where γ is the damping constant) and the sharp force $F(t) = s\delta(t - \sigma)$ of the individual kick occurring at time $t = \sigma$ (where s is the strength of the kick, and δ is the Dirac's 'delta' function). In this way, the (single) *kick equation* of the ball motion becomes

$$\dot{v} = -\gamma v(t) + s\delta(t - \sigma), \quad (2.149)$$

with the general solution

$$v(t) = sG(t - \sigma),$$

where $G(t - \sigma)$ is the *Green's function*³⁸

$$G(t - \sigma) = \begin{cases} 0 & \text{for } t < \sigma \\ e^{-\gamma(t-\sigma)} & \text{for } t \geq \sigma \end{cases}.$$

Now, we can generalize the above to N kicks with individual strengths s_j , occurring at a sequence of times $\{\sigma_j\}$, so that the total kicking force becomes

$$F(t) = \sum_{j=1}^N s_j \delta(t - \sigma_j).$$

³⁸ This is the Green's function of the first order system (2.149). Similarly, the Green's function

$$G(t - \sigma) = \begin{cases} 0 & \text{for } t < \sigma \\ (t - \sigma)e^{-\gamma(t-\sigma)} & \text{for } t \geq \sigma \end{cases}$$

corresponds to the second order system

$$\left(\frac{d}{dt} + \gamma\right)^2 G(t - \sigma) = \delta(t - \sigma).$$

In this way, we get the *multi-kick equation* of the ball motion

$$\dot{v} = -\gamma v(t) + \sum_{j=1}^N s_j \delta(t - \sigma_j),$$

with the general solution

$$v(t) = \sum_{j=1}^N s_j G(t - \sigma_j). \quad (2.150)$$

As a final generalization, we would imagine that the kicks are continuously exerted on the ball, so that kicking force becomes

$$F(t) = \int_{t_0}^T s(\sigma) \delta(t - \sigma) d\sigma \equiv \int_{t_0}^T d\sigma F(\sigma) \delta(t - \sigma),$$

so that the continuous multi-kick equation of the ball motion becomes

$$\dot{v} = -\gamma v(t) + \int_{t_0}^T s(\sigma) \delta(t - \sigma) d\sigma \equiv -\gamma v(t) + \int_{t_0}^T d\sigma F(\sigma) \delta(t - \sigma),$$

with the general solution

$$v(t) = \int_{t_0}^T d\sigma F(\sigma) G(t - \sigma) = \int_{t_0}^T d\sigma F(\sigma) e^{-\gamma(t-\sigma)}. \quad (2.151)$$

Random Kicks and Langevin Equation

We now denote the times at which kicks occur by t_j and indicate their direction in a one-dimensional game by $(\pm 1)_j$, where the choice of the plus or minus sign is random (e.g., throwing a coin). Thus the kicking force can be written in the form [Hak02]

$$F(t) = s \sum_{j=1}^N \delta(t - t_j) (\pm 1)_j, \quad (2.152)$$

where for simplicity we assume that all kicks have the same strength s . When we observe many games, then we may perform an average $\langle \dots \rangle$ over all these different *performances*,

$$\langle F(t) \rangle = s \langle \sum_{j=1}^N \delta(t - t_j) (\pm 1)_j \rangle. \quad (2.153)$$

Since the direction of the kicks is assumed to be independent of the time at which the kicks happen, we may split (2.153) into the product

$$\langle F(t) \rangle = s \langle \sum_{j=1}^N \delta(t - t_j) \rangle \langle (\pm 1)_j \rangle .$$

As the kicks are assumed to happen with equal frequency in both directions, we get the cancelation

$$\langle (\pm 1)_j \rangle = 0,$$

which implies that the average kicking force also vanishes,

$$\langle F(t) \rangle = 0.$$

In order to characterize the strength of the force (2.152), we consider a quadratic expression in F , e.g., by calculating the *correlation function* for two times t, t' ,

$$\langle F(t)F(t') \rangle = s^2 \langle \sum_j \delta(t - t_j)(\pm 1)_j \sum_k \delta(t' - t_k)(\pm 1)_k \rangle .$$

As the ones for $j \neq k$ will cancel each other and for $j = k$ will become 1, the correlation function becomes a single sum

$$\langle F(t)F(t') \rangle = s^2 \langle \sum_j \delta(t - t_j)\delta(t' - t_k) \rangle, \quad (2.154)$$

which is usually evaluated by assuming the *Poisson process* for the times of the kicks.

Now, proper description of random motion is given by *Langevin rate equation*, which describes the *Brownian motion*: when a particle is immersed in a fluid, the velocity of this particle is slowed down by a force proportional to its velocity and the particle undergoes a zig-zag motion (the particle is steadily pushed by much smaller particles of the liquid in a random way). In physical terminology, we deal with the behavior of a system (particle) which is coupled to a *heat bath* or reservoir (namely the liquid). The heat bath has two effects [Hak02]:

1. It decelerates the mean motion of the particle; and
2. It causes statistical fluctuation.

The standard Langevin equation has the form

$$\dot{v} = -\gamma v(t) + F(t), \quad (2.155)$$

where $F(t)$ is a *fluctuating force* with the following properties:

1. Its statistical average (2.153) vanishes; and
2. Its correlation function (2.154) is given by

$$\langle F(t)F(t') \rangle = Q\delta(t - t_0), \quad (2.156)$$

where $t_0 = T/N$ denotes the mean free time between kicks, and $Q = s^2/t_0$ is the *random fluctuation*.

The general solution of the Langevin equation (2.155) is given by (2.151).

The average velocity vanishes, $\langle v(t) \rangle = 0$, as both directions are possible and cancel each other. Using the integral solution (2.151) we get

$$\langle v(t)v(t') \rangle = \left\langle \int_{t_0}^t d\sigma \int_{t_0}^{t'} d\sigma' F(\sigma)F(\sigma')e^{-\gamma(t-\sigma)}e^{-\gamma(t'-\sigma')} \right\rangle,$$

which, in the steady-state, reduces to

$$\langle v(t)v(t') \rangle = \frac{Q}{2\gamma}e^{-\gamma(t-t')},$$

and for equal times

$$\langle v(t)^2 \rangle = \frac{Q}{2\gamma}.$$

If we now repeat all the steps performed so far with $m \neq 1$, the final result reads

$$\langle v(t)^2 \rangle = \frac{Q}{2\gamma m}. \quad (2.157)$$

Now, according to thermodynamics, the *mean kinetic energy* of a particle is given by

$$\frac{m}{2} \langle v(t)^2 \rangle = \frac{1}{2}k_B T, \quad (2.158)$$

where T is the (absolute) temperature, and k_B is the Boltzman's constant. Comparing (2.157) and (2.158), we get the important Einstein's result

$$Q = 2\gamma k_B T,$$

which says that whenever there is damping, i.e., $\gamma \neq 0$, then there are random fluctuations (or noise) Q . In other words, fluctuations or noise are inevitable in any physical system. For example, in a resistor (with the resistance R) the electric field E fluctuates with a correlation function (similar to (2.156))

$$\langle E(t)E(t') \rangle = 2Rk_B T \delta(t - t').$$

This is the simplest example of the *fluctuation-dissipation theorem*.

2.7.4 Mesoscopic Derivation of Order Parameters

The basic Hamiltonian equations (2.147–2.148) are in general quite complicated and can hardly be solved completely in the whole locomotor phase-space Γ , spanned by the set of possible joint vectors $\{q^i(t), p_i(t)\}$. We therefore have to restrict ourselves to local concepts for analyzing the behavior of our locomotor system. To this end we shall consider a reference musculo-skeletal state $\{q_0, p_0\}$ and its neighborhood. Following the procedures of the mesoscopic synergetics (see [Hak83, Hak93]), we assume that the reference state has the

properties of an attractor and is a comparably low-dimensional object in Γ . In order to explore the behavior of our locomotor system (dependent on the set of control parameters σ) in the neighborhood of $\{q_0, p_0\}$ we look for the time development of small deviations from the reference state (to make the formalism as simple as possible, we drop the joint index in this section)

$$q(t) = q_0 + \delta q(t), \quad p(t) = p_0 + \delta p(t),$$

and consider $\delta q(t)$ and $\delta p(t)$ as small entities. As a result we may linearize the equations of δq and δp in the vicinity of the reference state $\{q_0, p_0\}$. We get

$$\partial_t \delta q(t) = L[q_0, p_0, \sigma] \delta q(t), \quad \partial_t \delta p(t) = K[q_0, p_0, \sigma] \delta p(t),$$

where $L[\cdot]$ and $K[\cdot]$ are linear matrices independent of $\delta q(t)$ and $\delta p(t)$, which can be derived from the basic Hamiltonian vector-field (2.147–2.148) by standard synergetics methods [Hak83, Hak93, Hak96, Hak00]. We now assume that we can construct a complete set of eigenvectors $\{l^{(j)}(t), k^{(j)}(t)\}$ corresponding to (2.145). These eigenvectors allow us to decompose arbitrary deviations $\delta q(t)$ and $\delta p(t)$ into elementary collective deviations along the directions of the eigenvectors

$$\delta q(t) = \xi_j(t) l^j(t), \quad \delta p(t) = \zeta_j(t) k^j(t), \quad (2.159)$$

where $\xi_j(t)$ and $\zeta_j(t)$ represent the excitations of the system along the directions in the phase-space Γ prescribed by the eigenvectors $l^j(t)$ and $k^j(t)$, respectively. These amplitudes are still dependent on the set of control parameters $\{\sigma\}$. We note that the introduction of the eigenvectors $\{l^j(t), k^j(t)\}$ is of crucial importance. In the realm of synergetics they are considered as the *collective modes or patterns* of the system. Whereas the basic Hamiltonian equation (2.147–2.148) is formulated on the basis of the human locomotor-system variables (coordinates and momenta) of the single subsystems (joints), we can now give a new formulation which is based on these collective patterns and describes the dynamical behavior of the locomotor system in terms of these different collective patterns. Inserting relations (2.159) into the basic system (2.147–2.148) we get equations for the amplitudes $\xi_j(t)$ and $\zeta_j(t)$,

$$\dot{\xi}_i(t) = A_{ij} \cdot \xi_j(t) + \text{nonlinear terms}, \quad \dot{\zeta}_j(t) = B_{ij} \cdot \zeta_j(t) + \text{nonlinear terms},$$

where $\cdot$ denotes the scalar product, and it is assumed that the time dependence of the linear matrices L and K is carried out by the eigenvectors leaving us with constant matrices A and B .

We now summarize the results by discussing the following time-evolution formulas for joint coordinates $q(t)$ and momenta $p(t)$,

$$q(t) = q_0 + \xi_j(t) l^j(t), \quad p(t) = p_0 + \zeta_j(t) k^j(t), \quad (2.160)$$

which describes the time dependence of the phase vectors $q(t)$ and $p(t)$ through the evolution of the collective patterns. Obviously, the reference musculo-skeletal state $\{q_0(t), p_0(t)\}$ can be called stable when all the possible excitations $\{\xi_j(t), \zeta_j(t)\}$ decay during the course of time. When we now change

the control parameters $\{\sigma\}$ some of the $\{\xi_j(t), \zeta_j(t)\}$ can become unstable and start to grow in time. The border between decay and growth in parameter space is called a Tablecritical region. Haken has shown that the few unstable amplitudes, denoted by u_q and u_p , change very slowly in the vicinity of a critical region, whereas the damped amplitudes, denoted by s_q and s_p , quickly decay to values which are completely prescribed by the unstable modes. This fact is expressed as the Tableslaving principle of synergetics [Hak83, Hak93, Hak96, Hak00], in our case reading as

$$s_q = s_q(u_q), \quad s_p = s_p(u_p).$$

These relations allow us to eliminate the stable modes in (2.160), and leave us with a low-dimensional set of equations for the unstable modes which play the role of the order parameters. These Tableorder parameter equations then completely rule the behavior of our microscopic nD musculo-skeletal system on macroscopic scales near an instability.

The fundamental result of synergetics consists in the observation that on macroscopic scales new laws can be discovered which exist in their own right [Hak83]. These laws which are expressed by the order parameter equations turn out to be independent of the detailed nature of the subsystems and their interactions. As a consequence this allows us to introduce the concept of Tablenormal forms [Arn88] as a method to discuss instabilities and qualitative dynamics in the neighborhood of the critical regions. This method of phenomenological synergetics allows us to start qualitative analysis from purely macroscopic considerations.

Using the so-called Tableadiabatic elimination of fast variables [Hak83], one tries to identify macroscopic quantities related to global musculo-skeletal dynamics (similar but different from the *mean-field* center-of-mass dynamics) – from experience and classifies them according to time-scale arguments. The slowest variables are usually identified with the control parameters which are assumed to be quasi static quantities. The slow macroscopic dynamics of the system has to be attributed to the order parameters. Very quickly relaxing variables have to be considered as enslaved modes.

2.7.5 Example: Synergetic Control of Biodynamics

Recall from [II05, II06a, II06b] that the basic microscopic synergetic level of human musculo-skeletal dynamics (2.147–2.148), can be viewed on the highest, *macroscopic synergetic center-of-mass level* of human motion dynamics as a simple *Hamilton oscillator*, physically representing the damped, sinusoidally driven pendulum of the unit mass and length l

$$l^2 \ddot{q} + \gamma \dot{q} + lg \sin q = A \cos(p_D t).$$

This equation expresses Newtonian second law of motion with the various terms on the left representing acceleration, damping, and gravitation. The

angular momentum of the forcing p_D , may be different from the natural frequency of the pendulum. In order to minimize the number of adjustable parameters the equation may be rewritten in dimensionless form as

$$\ddot{q} + (1/\nu)\dot{q} + \sin q = \epsilon \cos(p_D t),$$

where ν is the damping or quality parameter, ϵ is the forcing amplitude, and p_D is the drive frequency. The low-amplitude natural angular frequency of the pendulum is unity, and time is regarded as dimensionless. This equation satisfies the necessary conditions for chaos when it is written as an extended Hamiltonian system

$$\dot{q} = p, \quad \dot{p} = -(1/\nu)p - \sin q + \epsilon \cos \phi, \quad \dot{\phi} = p_D. \quad (2.161)$$

The variable ϕ is introduced as the phase of the drive term. Three variables are evident and also two nonlinear coupling terms. Whether the motion is chaotic depends upon the values of the three parameters: damping, forcing amplitude and drive frequency. For some values the pendulum locks onto the driving force, oscillating in a periodic motion whose frequency is the driving frequency, possibly with some harmonics or subharmonics. But for other choices of the parameters the pendulum motion is chaotic. One may view the chaos as resulting from a subtle interplay between the tendency of the pendulum to oscillate at its ‘natural’ frequency and the action of the forcing term. The transitions between non-chaotic and chaotic states, due to changes in the parameters, occur in several ways and depend delicately upon the values of the parameters.

To include (in the simplest possible way) the muscle excitation-contraction dynamics, and thus make the damped, driven Hamilton oscillator (2.161) a more realistic macroscopic model for human motion dynamics, we assume that the time-dependent forcing amplitude $\epsilon = \epsilon(t)$ has the form of a low pass filter, a characteristic feature of biological systems, given by first-order transfer function $\frac{K}{Ts+1}$. Here K denotes gain of the filter and T its time constant.

Therefore, macroscopic mechanical model of human motion dynamics gets the fully-functional form

$$\ddot{q} + (1/\nu)\dot{q} + \sin q = K(1 - e^{-t/T}) \cos(p_D t),$$

which can be rewritten in the form of *extended Hamilton oscillator*

$$\dot{q} = p, \quad \dot{p} = -(1/\nu)p - \sin q + K(1 - e^{-t/T}) \cos \phi, \quad \dot{\phi} = p_D. \quad (2.162)$$

Now, to effectively control the macroscopic HML model (2.162), we can use two standard nonlinear-control techniques:

1. Adaptive Lie-derivative based geometric control; and
2. Adaptive fuzzy-logic based AI control.

2.7.6 Example: Chaotic Psychodynamics of Perception

Perceptual alternation phenomena of ambiguous figures have been studied for a long time. Figure-ground, perspective (depth) and semantic ambiguities are well known (see, e.g., [Att71, Hak91]). When we view the *Necker cube*, which is a classic example of perspective alternation, a part of the figure is perceived either as front or back of a cube and our perception switches between the two different interpretations (see Figure 2.70). In this circumstance the external stimulus is kept constant, but perception undergoes involuntary and random-like change. The measurements have been quantified in psychophysical experiments and it becomes evident that the times between such changes are approximately Gamma distributed [BMA72, BCR82, Hak91].

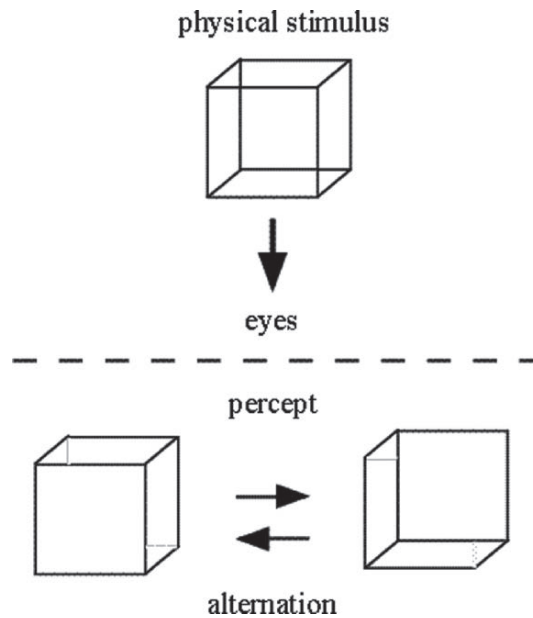


Fig. 2.70. Perception of the Necker cube with its two alternative interpretations (modified and adapted from [NNM00]).

Mathematical model approaches to explaining the facts have been made mainly from three situations based on the *synergetics* [DT89, DT90, CA93], the BSB (brain-state-in-a-box) neural network model [KA85, RMS90, MM95], and the PDP (parallel distributed processing) schema model [RM86, SKW95, IN96]. Common to these approaches is that top-down designs are applied so that the model can be manipulable by a few parameters and upon this basis fluctuating sources are brought in. The major interests seem to be not in the relation between the whole function and its element (neuron), but in the model building at the phenomenological level.

So far diverse types of chaotic dynamics have been confirmed at several hierarchical levels in the real neural systems from single cells to cortical networks (e.g., ionic channels, spike trains from cells, EEG) [Arb95].

Following [NNM00], in this section we present a perception model of ambiguous patterns based on the chaotic neural network from the viewpoint of bottom-up approach [NNM97], aiming at the functioning of chaos in dynamic perceptual processes.

The chaotic neural network (CNN) composed of N chaotic neurons is described as [ATT90, NKF97] (summation upon repeated indices is always understood)

$$X_i(t+1) = f(\eta_i(t+1) + \zeta_i(t+1)), \quad (2.163)$$

$$\eta_i(t+1) = w_{ij} \sum_{d=0}^t k_f^d X_j(t-d), \quad (2.164)$$

$$\zeta_i(t+1) = -\alpha \sum_{d=0}^t k_r^d X_i(t-d) - \theta_i, \quad (2.165)$$

where X_i : output of neuron i ($-1 \leq X_i \leq 1$), w_{ij} : synaptic weight from neuron j to neuron i , θ_i : threshold of neuron i , $k_f(k_r)$: decay factor for the feedback (refractoriness) ($0 \leq k_f, k_r < 1$), α : refractory scaling parameter, f : output function defined by $f(y) = \tanh(y/2\varepsilon)$ with the steepness parameter ε . Owing to the exponentially decaying form of the past influence, (2.164) and (2.165) can be reduced to

$$\eta_i(t+1) = k_f \eta_i(t) + w_{ij} X_j(t), \quad (2.166)$$

$$\zeta_i(t+1) = k_r \zeta_i(t) - \alpha X_i(t) + a, \quad (2.167)$$

where a is temporally constant $a \equiv -\theta_i(1 - k_r)$. All neurons are updated in parallel, that is, synchronously. The network corresponds to the conventional *Hopfield discrete-time network*:

$$X_i(t+1) = f[w_{ij} X_j(t) - \theta_i], \quad (2.168)$$

when $\alpha = k_f = k_r = 0$ (Hopfield network point (HNP)). The asymptotical stability and chaos in discrete-time neural networks are theoretically investigated in [MW89, CA97]. The stochastic fluctuation $\{F_i\}$ is attached to (2.168) of HNP together with the external stimulus $\{\sigma_i\}$:

$$X_i(t+1) = f[w_{ij} X_j(t) + \sigma_i + F_i(t)],$$

where $\begin{cases} \langle F_i(t) \rangle = 0 \\ \langle F_i(t) F_j(t') \rangle = D^2 \delta_{tt'} \delta_{ij}. \end{cases}$

Under external stimuli, (2.163) is influenced as

$$X_i(t+1) = f[\eta_i(t+1) + \zeta_i(t+1) + \sigma_i], \quad (2.169)$$

where $\{\sigma_i\}$ is the effective term by external stimuli. This is a simple and un-artificial incorporation of stimuli as the changes of neural active potentials.

The two competitive interpretations are embedded in the network as minima of the energy map (see Figure 2.71):

$$E = -\frac{1}{2}w_{ij}X_iX_j,$$

at HNP. This is done by using a iterative perception learning rule for $p(< N)$ patterns $\{\xi_i^\mu\} \equiv (\xi_1^\mu, \dots, \xi_N^\mu)$, $(\mu = 1, \dots, p; \xi_i^\mu = +1 \text{ or } -1)$ in the form :

$$w_{ij}^{new} = w_{ij}^{old} + \sum_{\mu} \delta w_{ij}^\mu, \quad \text{with} \quad \delta w_{ij}^\mu = \frac{1}{N} \theta(1 - \gamma_i^\mu) \xi_i^\mu \xi_j^\mu,$$

$$\text{where} \quad \gamma_i^\mu \equiv \xi_i^\mu w_{ij} \xi_j^\mu,$$

and $\theta(h)$ is the unit step function. The learning mode is separated from the performance mode by (2.169).

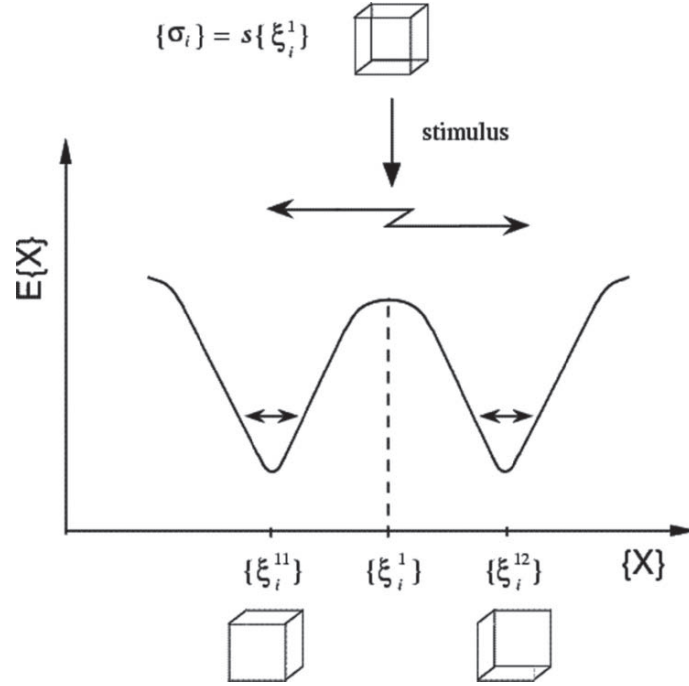


Fig. 2.71. Conceptual psychodynamic model of [NNM00], illustrating state transitions induced by chaotic activity.

Simulations of the CNN have shown that the neural chaos leads to perceptual alternations as responses to ambiguous stimuli in the chaotic neural

network. Its emergence is based on the simple process in a realistic bottom-up framework. In the same stage, similar results can not be obtained by the stochastic activity. This simulation suggests functional usefulness of the chaotic activity in perceptual systems even at higher cognitive levels. The perceptual alternation appears to be an inherent feature built in the chaotic neuron assembly. It may be interesting to study the brain with the experimental technique (e.g., functional MRI) under the circumstance where the perceptual alternation is running [NNM00].

Neuro–Dynamics, Synergetics and Synchronization

In this Chapter we develop our neurodynamics, neuro–synergetics and synchronization modeling approach to brain and cognition. Formally, here we are dealing with a *brain evolution* 2–functor $\mathcal{E}$, given by

$$\begin{array}{ccc}
 \begin{array}{ccc}
 A & \xrightarrow{f} & B \\
 \downarrow h & & \downarrow g \\
 C & \xrightarrow{k} & D
 \end{array} & \xrightarrow{\mathcal{E}} & \begin{array}{ccc}
 \mathcal{E}(A) & \xrightarrow{\mathcal{E}(f)} & \mathcal{E}(B) \\
 \downarrow \mathcal{E}(h) & & \downarrow \mathcal{E}(g) \\
 \mathcal{E}(C) & \xrightarrow{\mathcal{E}(k)} & \mathcal{E}(D)
 \end{array}
 \end{array}$$

CURRENT BRAIN STATE DESIRED BRAIN STATE

Here $\mathcal{E}$ represents an association/projection functor from the *source 2–category* of the current brain state, defined as a commutative square of small brain categories $A, B, C, D, \dots$ of *current brain components* and their causal interrelations $f, g, h, k, \dots$, onto the *target 2–category* of the desired brain state, defined as a commutative square of small brain categories $\mathcal{E}(A), \mathcal{E}(B), \mathcal{E}(C), \mathcal{E}(D), \dots$ of *evolved brain components* and their causal interrelations $\mathcal{E}(f), \mathcal{E}(g), \mathcal{E}(h), \mathcal{E}(k)$. As in the previous section, each causal arrow in above diagram, e.g., $f : A \rightarrow B$, stands for a *generic brain dynamorphism*.

3.1 Distinguished ANN Models

In this section we continue the exposition on ANNs, giving several distinguished ANN models that are used in modern brain dynamics (see [II06a] for details).

3.1.1 Generalized Kohonen’s SOM

As the brain is assumed to be optimized by evolution for information processing, one would postulate that maximal mutual information is a sound

principle governing the setup of neural structures. For feedforward neural structures with lateral inhibition, an algorithm of maximal mutual information has been defined by Linsker [Lin89] using the gradient descent in mutual information. It requires computationally costly integrations, and has a highly nonlocal learning rule and therefore is less favorable as a model for biological maps and less feasible for technical applications [CS02, Cla02a].

However, both biological network structures and technical applications are (due to realization constraints) not necessarily capable of reaching this optimum. This remains a question under discussion especially for the brain [Plu99]. Even if one had quantitative experimental measurements of the magnification behavior, the question from what self-organizing dynamics the neural structure emerged remains. So overall it is desirable to formulate other learning rules that minimize mutual information in a simpler way.

The self-organizing feature map algorithm was proposed in 1982 by Kohonen [Koh82] has become a successful model for topology preserving primary sensory processing in the cortex [OBS92], and an useful tool in technical applications [RMS92].

The *self-organizing feature maps* map an input space, such as the retina or skin receptor fields, into a neural layer by feedforward structures with lateral inhibition. Biological maps show as defining properties topology preservation, error tolerance, plasticity (the ability of adaptation to changes in input space), and self-organized formation by a local process, since the global structure cannot be coded genetically [CS02, Cla02a].

Compared to the *elastic net algorithm* of Durbin and Willshaw [DW87] and the Linsker algorithm [Lin89] which are performing gradient descent in a certain energy landscape, the Kohonen algorithm seems to have no energy function. Although the learning process can be described in terms of a Fokker-Planck equation [RS88], the expectation value of the learning step is a nonconservative force [OBS92] driving the process so that it has no associated energy function. Furthermore, the relationships of the Kohonen model to both alternative models and general principles are still an open field [Koh91].

The Winner Relaxing Kohonen Algorithm

We now consider an energy function V that was at first proposed in [RMS92] for the classical *Kohonen algorithm* (2.119). If we have a discrete input space, the potential function for the expectation value of the learning step is given by

$$V(\{\mathbf{w}\}) = \frac{1}{2} \sum_{\mathbf{r} \neq \mathbf{s}} g_{\mathbf{r}\mathbf{s}}^{\gamma} \sum_{\mu | \mathbf{v}^{\mu} \in F_{\mathbf{s}}(\{\mathbf{w}\})} p(\mathbf{v}^{\mu}) \cdot |\mathbf{v}^{\mu} - \mathbf{w}_{\mathbf{r}}|^2,$$

where $F_{\mathbf{s}}(\{\mathbf{w}\})$ is the cell of the *Voronoi tessellation* (or *Dirichlet tessellation*) of input space given by the weight vector configuration $\{\mathbf{w}\}$. The Voronoi tessellation is defined as the subset of vectors in input space that lead to firing of neuron $\mathbf{s}$. For discrete input space (where $p(\mathbf{v})$ is a sum over delta

peaks $\delta(\mathbf{v} - \mathbf{v}^\mu)$) the first derivative is not continuous at all weight vectors where the borders of the Voronoi tessellation are shifting over one of the input vectors.

Kohonen has, utilizing some approximations, shown in [Koh91] for the 1D or 2D case that a gradient descent in V – note that the borders of the Voronoi tessellation $F_{\mathbf{s}}(\{\mathbf{w}\})$ are shifting if one evaluates the gradient with respect to a weight vector $\mathbf{w}_{\mathbf{r}}$ – results in a slightly different learning rule, which differs from the classical Kohonen rule only for the winning neuron $\mathbf{s}$ itself:

$$\delta \mathbf{w}_{\mathbf{r}} = \eta \{ (\mathbf{v} - \mathbf{w}_{\mathbf{r}}) \cdot g_{\mathbf{rs}} - \frac{1}{2} \delta_{\mathbf{rs}} \sum_{\mathbf{r}' \neq \mathbf{s}} g_{\mathbf{r}'\mathbf{s}} (\mathbf{v} - \mathbf{w}_{\mathbf{r}'}) \}.$$

As the second term implies an additional elastic relaxation for the winning neuron, it is straightforward to call it ‘Winner Relaxing’ Kohonen algorithm. As the relaxing term acts only in one direction, the winner is relaxed to its neighbors, but the neighbors stay unattracted, it can not strictly be interpreted as an elastic force or physical interaction.

The Magnification Factor

Depending on the input probability density $P(\mathbf{v})$ of the stimuli, any self-organizing map algorithm should show the property to spend more neurons to represent areas of higher probability density, according to a higher resolution, which is quantified by the magnification factor.

The magnification factor is defined as the density of neurons $\mathbf{r}$ (i.e., the density of synaptic weight vectors $\mathbf{w}_{\mathbf{r}}$) per unit volume of input space, and therefore is given by the inverse Jacobian of the mapping from input space to neuron layer: $M = |J|^{-1} = |\det(d\mathbf{w}/d\mathbf{r})|^{-1}$. (In the following we consider the case of non-inverting mappings, where J is positive. Further we assume the input space now to be continuous and of same dimension as the neural layer.) The magnification factor is a property of the networks’ response to a given probability density of stimuli $P(\mathbf{v})$. To evaluate M in higher dimensions, one in general has to compute the equilibrium state of the whole network and needs therefore the complete global knowledge on $P(\mathbf{v})$ [CS02, Cla02a].

For 1D maps (and possibly for special separable but therefore less general cases in higher dimensions) the magnification factor can follow an universal magnification law, that is, $M(\bar{\mathbf{w}}(\mathbf{r}))$ is a function of the local probability density P only and is independent of both the location $\mathbf{r}$ in the neural layer and the location $\bar{\mathbf{w}}(\mathbf{r})$ in input space.

For the classical Kohonen algorithm the magnification law (for 1D maps) is given by a power law $M(\bar{\mathbf{w}}(\mathbf{r})) \propto P(\bar{\mathbf{w}}(\mathbf{r}))^\rho$ with exponent $\rho = \frac{2}{3}$ [RS86].

Further it is nontrivial whether there exists a power law or not, as in the case of the Elastic Net, which has an universal magnification law which however is not a power law [CS02].

An optimal map from the view of information theory would reproduce the input probability exactly ($M \sim P(\mathbf{v})^\rho$ with $\rho = 1$), according to a power law

with exponent 1. This is equivalent to the condition that all neurons in the layer are firing with same probability. An exponent $\rho = 0$, on the other hand, corresponds to a uniform distribution of weight vectors, which means there is no adaptation to the stimuli at all. So the magnification exponent is a direct indicator, how far a Self Organizing Map algorithm is away from the optimum predicted by information theory.

In the following subsection, the *Winner-relaxing Kohonen algorithm* (WRKA) is generalized introducing an additional parameter, and the magnification law in the 1D case is derived. This generalization can be used to pre-select the magnification exponent between 1/2 and 1 by a fixed choice of the parameter.

Magnification Exponent

The necessary condition for the final state of the algorithm is that for all neurons $\mathbf{r}$ the expectation value of the learning step vanishes:

$$\int d\mathbf{v} p(\mathbf{v}) \delta \mathbf{w}_{\mathbf{r}}(\mathbf{v}) = 0 \quad \text{for all } \mathbf{r} \in \mathbb{R}. \quad (3.1)$$

This is the *Chapman-Kolmogorov equation* for the *stochastic learning process* of serial presentation. Since this expectation value is equal to the learning step of the pattern parallel rule, (3.1) is the stationary state condition for *both* serial and parallel updating, and also for batch updating, so we can proceed for these variants simultaneously [CS02, Cla02a].

Now we derive the magnification law for WRKA. We first introduce the following generalization, referred to as generalized WRKA, with free parameters λ and μ :

$$\delta \mathbf{w}_{\mathbf{r}} = \eta \{ (\mathbf{v} - \mathbf{w}_{\mathbf{r}}) g_{\mathbf{r}\mathbf{s}}^{\gamma} + \mu (\mathbf{v} - \mathbf{w}_{\mathbf{r}}) \delta_{\mathbf{r}\mathbf{s}} - \lambda \delta_{\mathbf{r}\mathbf{s}} \sum_{\mathbf{r}' \neq \mathbf{s}} g_{\mathbf{r}'\mathbf{s}}^{\gamma} (\mathbf{v} - \mathbf{w}_{\mathbf{r}'}) \},$$

where $\mathbf{s}$ is the center of excitation for incoming stimulus $\mathbf{v}$, and $g_{\mathbf{r}\mathbf{s}}^{\gamma}$ is a Gaussian function of distance in the neural layer with characteristic length γ . The original algorithm proposed by Kohonen in 1991 is obtained for $\lambda = +1/2$ and $\mu = 0$, whereas the classical SOM algorithm is obtained for $\lambda = 0$ and $\mu = 0$. Only for the special case $\lambda = +1/2$ and $\mu = 0$ the algorithm is associated with the potential function.

Insertion of the update rule (3.2) into the stationarity condition: for all $\mathbf{r}$, $\int d\mathbf{v} P(\mathbf{v}) \cdot \delta \mathbf{w}_{\mathbf{r}} = 0$, and integration yields the differential equation (with $\bar{P} = P(\bar{w}(r))$),

$$\gamma^2 \left(J \frac{d(\bar{P}J)}{dr} + \frac{\bar{P}J}{2} \frac{dJ}{dr} + \lambda \frac{\bar{P}J}{2} \frac{dJ}{dr} \right) = 0.$$

For $\gamma \neq 0$, $P \neq 0$, $d\bar{P}/dr \neq 0$ and making the ansatz $J(r) = J(\bar{P}(r))$ of an universal local magnification law (that may be expected for the 1D case) we get the differential equation

$$\frac{dJ}{d\bar{P}} = -\frac{2}{3+\lambda} \frac{J}{\bar{P}},$$

with its solution (provided that $\lambda \neq -3$):

$$M = \frac{1}{J} \sim P(v)^{\frac{2}{3+\lambda}}.$$

For WRKA ($\lambda = 1/2$) the magnification factor follows an exact power law with magnification exponent $\rho = 4/7$, which is smaller than $\rho = 2/3$ [RS86] for the classical SOM. Although WRKA was reported as ‘somewhat faster’ [Koh91] in the initial ordering process, the resulting invariant mapping is slightly less optimal in terms of information theory [CS02, Cla02a].

3.1.2 Dynamics of Hopfield’s Associative Recurrent Nets

Ising–Spin Neurons

The paradigm for the unsupervised, self-organizing, associative, and recurrent ANN is the discrete Hopfield network (see [Hop82]). Hopfield gives a collection of simple threshold automata, called *formal neurons* by McCulloch and Pitts (see [Hay94]): two-state, ‘all-or-none’, firing or non-firing units that can be modeled by *Ising spins* (uniaxial magnets) $\{S_i\}$ such that $S_i = \pm 1$ (where $1 = |\uparrow\rangle$ = ‘spin up’ and $-1 = |\downarrow\rangle$ = ‘spin down’; the label of the neuron is i and ranges between 1 and the size of the network N). The neurons are connected by synapses J_{ij} .

Firing *patterns* $\{\xi_i^\mu\}$ represent specific S_i –*spin configurations*, where the label of the pattern is μ and ranges between 1 and q .

Using random patterns $\xi_i^\mu = \pm 1$ with equal probability $1/2$, we have the *synaptic efficacy* J_{ij} of j th neuron operating on i th neuron given by

$$J_{ij} = N^{-1} \xi_i^\mu \xi_j^\mu \equiv N^{-1} \boldsymbol{\xi}_i \cdot \boldsymbol{\xi}_j. \quad (3.2)$$

Postsynaptic potential (PSP) represents an *internal local field*

$$h_i(t) = J_{ij} S_j(t). \quad (3.3)$$

Now, the *sequential (threshold) dynamics* is defined in the form of discrete equation

$$S_i(t + \Delta t) = \text{sgn}[h_i(t)]. \quad (3.4)$$

Dynamics (3.4) is equivalent to the rule that the state of a neuron is changed, or a spin is flipped iff the total network *energy*, given by *Ising Hamiltonian*

$$H_N = -\frac{1}{2} J_{ij} S_i S_j, \quad (3.5)$$

is lowered [Hop82, HT85]. Therefore, the Ising Hamiltonian H_N represents the monotonically decreasing *Lyapunov function* for the sequential dynamics

(3.4), which converges to a local minimum or ground state of H_N . This holds for any *symmetric coupling* $J_{ij} = J_{ji}$ with $J_{ii} = 0$ and if spin-updating in (3.4) is asynchronous. In this case the patterns $\{\xi_i^\mu\}$ after convergence become identical, or very near to, ground states of H_N , each of them at the bottom of the valley.

Data are *stored* in the neural net if, by a suitable choice of the J_{ij} , several specific patterns $\{\xi_i^\mu\}$ are made local minima of H_N . If this can be achieved, the neural net will function as *content-addressable* or (auto)*associative memory*. A network state which 'somehow resembles' one of the stored prototypes corresponds to a location in the energy landscape which is close enough to the minimum representing that prototype to lie in its *basin of attraction*. By spontaneously moving downhill, or relaxing to the energy minimum, the network *recalls* the data or reconstructs the prototype.

Suppose that we have somehow stored several (stationary) patterns $\{\xi_i^\mu\}$ in the J_{ij} and that the system is offered a noisy version of one of them. If the noise was not too strong, the system remains in the valley associated with that pattern and under its natural dynamics it will relax to the energy minimum where the stored patterns live. That is, the system has recalled the pattern.

In statistical mechanics, one is usually given the synapses J_{ij} and one of the first tasks consists in finding the minima of the Ising Hamiltonian H_N (3.92). In the theory of neural networks, however, one is given the patterns $\{\xi_i^\mu\}$ and one is asked to solve the *inverse problem*: finding synapses J_{ij} such that the patterns $\{\xi_i^\mu\}$ are minima of the Hamiltonian H_N .

To see why the Hopfield model with synapses given by (3.2) has patterns $\{\xi_i^\mu\}$ as *attractors* of the dynamics (3.4), note that the sequential dynamical law embodies a two-step process, the evolution of the local field (PSP) (3.90), which is a *linear* operation, and a *nonlinear* decision process (3.4).

Graded-Response Neurons

Graded-response neurons have continuous input-output relation (like nonlinear operational amplifiers) of the form $V_i = g_i(\lambda u_i)$, where u_i denotes the input at i , a constant λ is called the gain parameter, and V_i is the output [Hop84]. Usually, g_i are taken to be sigmoid functions, odd, and monotonically increasing (e.g., $g(\cdot) = \frac{1}{2}(1 + \tanh(\cdot))$), while discrete Ising spins have $g_i(u_i) = \text{sgn}_i(u_i)$. The behavior of the *continuous Hopfield network* is usually described by a set of coupled RC-transient equations

$$C_i \dot{u}_i = I_i + J_{ij} V_j - \frac{u_i}{R_i}, \quad (3.6)$$

where $u_i = g^{-1}(V_i)$, R_i and C_i denote input capacitance and resistance, and I_i represents an external source.

The Hamiltonian of the continuous system (3.6) is given by

$$H = -\frac{1}{2} J_{ij} V_i V_j + \sum_{i=1}^N R_i^{-1} \int_0^{V_i} dV g^{-1}(V) - I_i V_i. \quad (3.7)$$

However, according to Hopfield [Hop84] the synapses J_{ij} retain the form (3.2) with random patterns $\xi_i^\mu = \pm 1$ with equal probability $1/2$, and the synaptic symmetry $J_{ij} = J_{ji}$ implies that the continuous Hamiltonian (3.7) represents a *Lyapunov function* of the system (3.6), i.e., H decreases under the continual neuro-dynamics governed by equation (3.6) as time proceeds.

More general form of synapses is

$$J_{ij} = N^{-1}Q(\xi_i; \xi_j),$$

for some synaptic kernel Q on $\mathbb{R}^n \times \mathbb{R}^n$. The vector ξ_i varies as i travels from 1 to N , but remains on a corner of the *Hamming hypercube* $[-1, 1]^q$.

Hopfield's Overlaps

Assuming that the number q of stored patterns is small compared to the number of neurons, i.e., $q/N \rightarrow 0$, we find that the synapses (3.2) give rise to a local field of the form

$$h_i = \xi_i^\mu m_\mu, \quad \text{where} \quad (3.8)$$

$$m_\mu = N^{-1} \xi_i^\mu S_i \quad (3.9)$$

is the *auto-overlap* (or simply *overlap*)¹ of the network state $\{S_i\}$ with the pattern $\{\xi_i^\mu\}$, measuring the proximity between them. We can see that $m_\mu = 1$ (like peak-up in auto-correlation) if $\{S_i\}$ and $\{\xi_i^\mu\}$ are identical patterns, $m_\mu = -1$ (like peak-down in autocorrelation) if they are each other's complement, and $m_\mu = O(1/\sqrt{N})$ if they are uncorrelated (like no-peak in auto-correlation) with each other. Overlaps m_μ are related to the Hamming distance d_μ between the patterns (the fraction of spins which differ) by $d_\mu = \frac{1}{2}(1 - m_\mu)$.

As a pattern ξ_i^μ represents (in the simplest case) a specific Ising-spin S_i -configuration, then $(\xi_i^\mu)^2 = 1$. If $S_i = \xi_i^\mu$ for all i , then $m_\mu = 1$. Conversely, if $m_\mu = 1$, then $S_i = \xi_i^\mu$. In all other cases $m_\mu < 1$, by the Cauchy-Schwartz inequality. If ξ_i^μ and S_i are uncorrelated, we may expect m_μ to be of the order of $N^{-1/2}$, since the sum consists of N terms, each containing a ξ_i^μ . On the other hand, if the S_i are positively correlated with ξ_i^μ , then m_μ is of the order of unity. So the overlaps give the global information about the network and hence are good order parameters. Also, according to Hopfield [Hop84], the extension to the continual network is straightforward.

Using overlaps, the *Ising Hamiltonian* becomes

$$H_N = -\frac{1}{2}N \sum_{\mu=1}^q m_\mu^2. \quad (3.10)$$

¹ resembling the auto-correlation function of a time-series, where distinct peaks indicate that the series at the certain time t is similar to the series at time $t + \Delta t$

The similarity between two different patterns ξ_i^μ and ξ_i^ν is measured by their *mutual overlap* or *cross-overlap* $m_{\mu\nu}$ (in other parlance it is called *Karhunen-Loeve covariance matrix* (see [FS92]), which extracts the principal components from a data set)², equal

$$m_{\mu\nu} = N^{-1} \xi_i^\mu \xi_i^\nu. \quad (3.11)$$

For similar patterns the cross-overlap is close to unity whereas for uncorrelated patterns it is random variable with zero mean and small $(1/\sqrt{N})$ variance.

The symmetric *Hopfield synaptic matrix* J_{ij} can be expressed in terms of the cross-overlaps $m_{\mu\nu}$ as

$$J_{ij} = N^{-1} \xi_i^\mu (m_{\mu\nu})^{-1} \xi_j^\nu = J_{ji}, \quad (3.12)$$

where $(m_{\mu\nu})^{-1}$ denotes the *Moore-Penrose pseudoinverse* of the cross-overlap matrix $m_{\mu\nu}$.

Besides the Hopfield model, the proposed pattern-overlap picture can be extended to cover some more sophisticated kinds of associative memory, among them (see, e.g., [DHS91]):

1. Forgetful memories, characterized by iterative synaptic prescription

$$J_{ij}^{(\mu)} = \phi(\epsilon \xi_i^\mu \xi_j^\mu + J_{ij}^{(\mu-1)}),$$

for some small parameter ϵ and some odd function ϕ . If $\phi(\cdot)$ saturates as $|\cdot| \rightarrow \infty$, the memory creates storage capacity for new patterns by forgetting the old ones.

2. Temporal associative memories, which can store and retrieve a sequence of patterns, through synapses

$$N J_{ij} = \xi_i^\mu \xi_j^\mu + \epsilon \sum_{\mu=1}^q \xi_i^{(\mu+1)} \xi_j^\mu, \quad (3.13)$$

where the second term on the right is associated with a temporal delay, so that one can imagine that the second term ‘pushes’ the neural system through an energy landscape created by the first term.

Overlap Dynamics

According to Hopfield [Hop84], the extension of the sequential dynamics $S_i = \text{sgn}(\sum_\mu m_{\mu} \xi_i^\mu)$ of the network made of the simplest Ising-spin-neurons to the

² resembling the cross-correlation function of two time-series, with several distinct peaks, indicating that the two series are very similar at each point in time where the peaks occur

network made of continual graded-response amplifier-neurons, is straightforward using the probabilistic *Glauber dynamics*

$$\text{Prob}\{S_i \mapsto -S_i\} = \frac{1}{2}[1 - \tanh(\beta h_i S_i)], \quad i = 1, \dots, N, \quad (3.14)$$

where β represents the universal temperature ($\beta = \frac{1}{k_B T}$, k_B is the normalized Boltzman's constant and $k_B T$ has dimension of energy).

Under the Glauber's dynamics (3.14), and as $N \rightarrow \infty$ (transition from the single neurons to the neural field), for time-dependent patterns $\xi^\mu(t) = \xi_\mu(t)$, vector auto-overlaps $m_\mu(t)$, and tensor cross-overlaps $m_{\mu\nu}(t)$, we present the dynamics of overlaps governed by the following nonlinear differential equations (generalized from [DHS91], pp. 23), respectively in the vector form

$$\dot{m}_\mu(t) = -m_\mu(t) + \langle \xi_\mu(t) \tanh[\beta m_\mu(t) \xi^\mu(t)] \rangle, \quad (3.15)$$

and in the tensor form

$$\dot{m}_{\mu\nu}(t) = -m_{\mu\nu}(t) + \langle \xi_\mu(t) \xi_\nu(t) \tanh[\beta m_{\mu\nu}(t) \xi^\mu(t) \xi^\nu(t)] \rangle, \quad (3.16)$$

where the angular brackets denote an average over the q patterns $\xi^\mu(t)$.

The stationary solutions (for any fixed instant of time $t = \tau$) of equations (3.15) and (3.16) are given by corresponding fixed-point vector and tensor equations

$$m_\mu = \langle \xi_\mu \tanh[\beta m_\mu \xi^\mu] \rangle, \quad \text{and} \quad (3.17)$$

$$m_{\mu\nu} = \langle \xi_\mu \xi_\nu \tanh[\beta m_{\mu\nu} \xi^\mu \xi^\nu] \rangle, \quad (3.18)$$

respectively.

Hebbian Learning Dynamics

In terms of stochastic feed-forward multi-layer neural networks, the tensorial equation (3.16) corresponds to the average, general, self-organizing Hebbian neural learning scheme (see [Heb49, Kos92])

$$\dot{m}_{\mu\nu}(t) = -m_{\mu\nu}(t) + \langle \mathcal{I}_{\mu\nu} \rangle, \quad (3.19)$$

with random signal *Hebbian innovation*

$$\mathcal{I}_{\mu\nu} = f_\mu[\xi^\mu(t)] f_\nu[\xi^\nu(t)] + \sigma_{\mu\nu}(t), \quad (3.20)$$

where $\sigma_{\mu\nu}$, denotes the tensorial, additive, zero-mean, Gaussian white-noise, independent of the main innovation function $I_{\mu\nu}$, while $f_{\mu,\nu}[\cdot]$ represent the hyperbolic tangent (sigmoid) neural activation functions. A single-layer Hebbian learning scheme, corresponding to the tensor equation (3.19), gives

$$\dot{m}_\mu(t) = -m_\mu(t) + \langle \mathcal{I}_\mu \rangle, \quad (3.21)$$

with the vector innovation

$$\mathcal{I}_\mu = f_\mu[\xi^\mu(t)] + \sigma_\mu(t),$$

where σ_μ , denotes the vector additive zero-mean Gaussian white-noise, also independent of the main innovation function I_μ , while $f_\mu[\cdot]$ represents the hyperbolic tangent (sigmoid) neural activation function.

If we assume the small absolute value of the average (stochastic) terms, the nonlinear overlap-dynamics equations (3.15) and (3.16) can be presented in the form of *weakly-connected neural networks* (see [HI97]), respectively, as a single-layer network

$$\dot{m}_\mu(t) = -m_\mu(t) + \varepsilon g_\mu(m_\mu, \varepsilon), \quad \varepsilon \ll 1, \quad (3.22)$$

and a multi-layer network

$$\dot{m}_{\mu\nu}(t) = -m_{\mu\nu}(t) + \varepsilon g_{\mu\nu}(m_{\mu\nu}, \varepsilon), \quad \varepsilon \ll 1, \quad (3.23)$$

where, g_μ and $g_{\mu\nu}$, corresponding to the average (bracket) terms in (3.15) and (3.16), describe (vector and tensor, respectively) synaptic connections and the ‘small’ parameter ε describes their (dimensionless) strength. These weakly-connected neural systems represent ε -perturbations of the corresponding linear systems

$$\dot{m}_\mu(t) = -m_\mu(t) \quad \text{and} \quad \dot{m}_{\mu\nu}(t) = -m_{\mu\nu}(t),$$

with exponential maps as solutions

$$m_\mu(t) = m_\mu e^{-t} \quad \text{and} \quad m_{\mu\nu}(t) = m_{\mu\nu} e^{-t},$$

using the stationary (fixed-point) solutions (3.17, 3.18) as initial conditions m_μ and $m_{\mu\nu}$. According to the *Hartman-Grobman theorem* from dynamical systems theory, the weakly-connected systems (3.22, 3.23) are topologically equivalent (homeomorphic) to the corresponding linear systems. Therefore the whole analysis for the linear vector and matrix flows can be applied here, with only difference that instead of increasing transients e^t here we have decreasing (i.e., asymptotically-stable) transients e^{-t} .

On the other hand, in terms of *synergetics* [Hak83], both nonlinear overlap-dynamics equations (3.15–3.16) and Hebbian learning equations (3.19–3.20), represent (covariant) *order parameter equations*.

By introducing the scalar quadratic potential fields, dependent on vector and tensor order parameters (overlaps), respectively

$$V(m_\mu) = -\frac{1}{2} \sum_{\mu=1}^q m_\mu^2 \quad \text{and} \quad V(m_{\mu\nu}) = -\frac{1}{2} m_{\mu\nu}^2,$$

we can generalize the overlap-dynamics equations (3.15–3.16) to the *stochastic-gradient order parameter equations*, in vector and tensor form, respectively

$$\dot{m}_\mu(t) = -\frac{\partial V(m_\mu)}{\partial m_\mu(t)} + F_\mu(t), \quad (3.24)$$

and

$$\dot{m}_{\mu\nu}(t) = -\frac{\partial V(m_{\mu\nu})}{\partial m_{\mu\nu}(t)} + F_{\mu\nu}(t). \quad (3.25)$$

$F_\mu(t)$ in (3.24) represents a vector fluctuating force, with average (over the stochastic process which produces the fluctuating force $F_\mu(t)$)

$$\langle F_\mu(t) \rangle = \langle \xi_\mu(t) \tanh[\beta m_\mu(t) \xi^\mu(t)] \rangle,$$

and variation

$$\langle F_\mu(t) F_\mu(t') \rangle = Q_\mu \delta(t - t'), \quad (3.26)$$

while $F_{\mu\nu}(t)$ in (3.25) represents a tensor fluctuating force, with average (over the stochastic process which produces the fluctuating force $F_{\mu\nu}(t)$)

$$\langle F_{\mu\nu}(t) \rangle = \langle \xi_\mu(t) \xi_\nu(t) \tanh[\beta m_{\mu\nu}(t) \xi^\mu(t) \xi^\nu(t)] \rangle,$$

and variation

$$\langle F_{\mu\nu}(t) F_{\mu\nu}(t') \rangle = Q_{\mu\nu} \delta(t - t'). \quad (3.27)$$

Coefficients Q_μ in (3.26) and $Q_{\mu\nu}$ in (3.27) represent strengths of the corresponding stochastic processes, while Dirac δ -functions $\delta(t - t')$ express their short-term memories.

Recall that standard interpretation of synergetics (see [Hak83]) describes the stochastic gradient systems (3.24–3.25) as the overdamped motion of (vector and tensor, respectively) representative particles in scalar potential fields $V(m_\mu)$ and $V(m_{\mu\nu})$, subject to fluctuating forces $F_\mu(t)$ and $F_{\mu\nu}(t)$. These particles undergo *non-equilibrium phase transitions* (in the similar way as the magnet undergoes transition from its unmagnetized state into a magnetized state, or a superconductor goes from its normal state into the superconducting state, only occurring now in systems far from thermal equilibrium), and associated phenomena, including a *symmetry breaking instability* and *critical slowing down* (see [Hak83]).

The non-equilibrium phase transitions of vector and tensor order parameters (overlaps) $m_\mu(t)$ and $m_{\mu\nu}(t)$, are in synergetics described in terms of probability distributions $p(m_\mu, t)$ and $p(m_{\mu\nu}, t)$, respectively, defined by corresponding *Fokker–Planck equations*

$$\dot{p}(m_\mu, t) = p(m_\mu, t) + \frac{1}{2} Q_\mu \frac{\partial^2 p(m_\mu, t)}{\partial m_\mu^2},$$

and

$$\dot{p}(m_{\mu\nu}, t) = p(m_{\mu\nu}, t) + \frac{1}{2} Q_{\mu\nu} \frac{\partial^2 p(m_{\mu\nu}, t)}{\partial m_{\mu\nu}^2}.$$

3.1.3 A Self-Organizing Bidirectional Competitive Net

Hopfield recurrent associative memory network can be generalized to get a *bidirectional associative memory* network, the so-called BAM model of Kosko [Kos92]. Here we derive an alternative self-organizing neural net model with competitive *Volterra-Lotka ensemble dynamics* [Ahm93].

We start from $(n + m)$ D linear ODEs, describing two competitive neural ensembles participating in a two-party game,

$$\begin{aligned}\dot{R}^i &= -\alpha_B^j B_j, & R^i(0) &= R_0^i, \\ \dot{B}_j &= -\beta_i^R R^i, & B_j(0) &= B_j^0, \quad (i = 1, \dots, n; j = 1, \dots, m),\end{aligned}\quad (3.28)$$

where $R^i = R^i(t)$ and $B_j = B_j(t)$ respectively represent the numerical strengths of the two neural ensembles at time t , R_0^i, B_j^0 are their initial conditions, and α_B and β^R represent the effective spiking rates (which are either constant, or Poisson random process). In this way, we generate a $(n + m)$ D smooth manifold M , a *neural state-space*, and two dynamical objects acting on it: an n D smooth *vector-field* $\dot{R}^i$, and an m D differential 1-form $\dot{B}_j$. Their dot product $\dot{R}^i \cdot \dot{B}_j$, represents a hypothetical *neural outcome*. This is a linear system, with the passive-decay couplings $\alpha_B^j B_j$ and $\beta_i^R R^i$, fully predictable but giving only equilibrium solutions.

Secondly, to incorporate competitive dynamics of Volterra-Lotka style as commonly used in ecological modeling and known to produce a global chaotic attractor [Ahm93], we include to each of the neural ensembles a nonlinear competing term depending only on its own units,

$$\begin{aligned}\dot{R}^i &= a^i R^i(1 - b^i R^i) - \alpha_B^j B_j, \\ \dot{B}_j &= c_j B_j(1 - d_j B_j) - \beta_i^R R^i.\end{aligned}\quad (3.29)$$

Now we have a *competition between the two chaotic attractors*, one for the R^i and one for the B_j ensemble, i.e., the two self-organization patterns emerging far-from-equilibrium.

Thirdly, to make this even more realistic, we include the ever-present *noise* in the form of Langevin-type random forces $F^i = F^i(t)$, and $G_j = G_j(t)$, thus adding the ‘neural heating’, i.e., noise induced entropy growth, to the competitive dynamics

$$\begin{aligned}\dot{R}^i &= a^i R^i(1 - b^i R^i) - \alpha_B^j B_j + F^i, \\ \dot{B}_j &= c_j B_j(1 - d_j B_j) - \beta_i^R R^i + G_j.\end{aligned}\quad (3.30)$$

Finally, to overcome the deterministic chaos and stochastic noise with an adaptive *brain-like dynamics*, we introduce the *field competition potential* V , in the scalar form

$$V = -\frac{1}{2}(\omega_i^j R^i B_j + \varepsilon_i^j B_j R^i), \quad (3.31)$$

where ω_i^j and ε_i^j represent *synaptic associative-memory* matrices for the R^i and B_j ensemble, respectively. From the negative potential V , we get a *Lyapunov-stable gradient system* $\dot{R}^i = -\frac{\partial V}{\partial B_j}$, $\dot{B}_j = -\frac{\partial V}{\partial R^i}$. This robust system, together with the *sigmoidal activation functions* $S(\cdot) = \tanh(\cdot)$, and *control inputs* $u_{OLN}^i = u_{OLN}^i(t)$ and $v_j^{OLN} = v_j^{OLN}(t)$, we incorporate into (3.30) to get the *full neural competitive-associative dynamics* (compare with [IJB99a])

$$\begin{aligned}\dot{R}^i &= u_{OLN}^i - \alpha_B^j B_j + a^i R^i (1 - b^i R^i) + \omega_i^j S_j(B_j) + F^i, \\ \dot{B}_j &= v_j^{OLN} - \beta_i^R R^i + c_j B_j (1 - d_j B_j) + \varepsilon_i^j S^i(R^i) + G_j, \\ &\text{with initial conditions} \\ R^i(0) &= R_0^i, \quad B_j(0) = B_j^0.\end{aligned}\quad (3.32)$$

Now, each ensemble learns by trial-and-error from the opposite side. In a standard ANN-fashion, we model this learning on the spot by initially setting the random values to the synaptic matrices ω_i^j and ε_i^j , and subsequently adjust these values using the standard *Hebbian learning scheme*:

New Value = Old Value + Innovation. In our case it reads:

$$\begin{aligned}\dot{\omega}_i^j &= -\omega_i^j + \Phi_i^j(R^i, B_j), \\ \dot{\varepsilon}_i^j &= -\varepsilon_i^j + \Psi_i^j(B_j, R^i),\end{aligned}\quad (3.33)$$

with *innovation* given in tensor signal form (generalized from [Kos92])

$$\begin{aligned}\Phi_i^j &= S_j(R^i) S_j(B_j) + \dot{S}_j(R^i) \dot{S}_j(B_j), \\ \Psi_i^j &= S^i(R^i) S^i(B_j) + \dot{S}^i(R^i) \dot{S}^i(B_j),\end{aligned}\quad (3.34)$$

where terms with overdots, equal $\dot{S}(\cdot) = 1 - \tanh(\cdot)$, denote the *signal velocities*.

3.1.4 Vector AM-Neural Nets

Recall that the number of patterns that can be stored in the *Hopfield model* (HM) is comparatively not very large. If N is the number of binary neurons, then the thermodynamic approach leads to the well-known estimation of the HM storage capacity, $p_{HM} \sim 0.14 \cdot N$ ([AGS95], [HKP93]). At the early 90s some authors suggested Hopfield-like models of associative memory with q -valued neurons that could be in more than two different states, $q \geq 2$, [Kan88]. All these models are related to the magnetic *Potts model*, which generalizes the *Ising-spin-glass model* for the spin variable that takes $q > 2$ different values [Wu82, Bax82]. In all these works, the authors used the same well-known approach linking the Ising model to the Hopfield model (see, for example, [HKP93]). In this approach, in place of the short-range interaction between two nearest spins the Hebbian interconnections between all q -valued

neurons were used. As a result, the long-range interactions appear. Then in the mean-field approximation it was possible to calculate the statistical sum and, consequently, to construct the phase diagram. Different regions of the phase diagram were interpreted in terms of the network ability to recognize noisy patterns.

For all these models, except one, the storage capacity is even less than that for HM. An exception is termed the *Potts-glass neural network* (PGNN) [Kan88]. The numerical solution of transcendental equations resulting from this thermodynamic approach leads to the following estimation for storage capacity for PGNN

$$p_{PGNN} \sim \frac{q(q-1)}{2} \cdot p_{HM}.$$

As far as q -valued models are intended for color images processing, number q stands for number of different colors, used for elementary pixel can be painted. Even if $q \sim 10$ the storage capacity of PGNN is 50 times as much as the storage capacity of HM. For computer processing of colored images the standard value is $q = 256$. Consequently, comparing with HM the gain is about four orders, $p_{PGNN} \sim 10^4 \cdot p_{HM}$, which is very good result. However, for long time it has not been clear why PGNN has such a big storage capacity, as thermodynamic approach does not answer this question.

On the other hand, recently [KM02, FK01] worked out the related AM model, intended for implementation as an optical device. Such a network is capable to hold and handle information that is encoded in the form of the *frequency-phase modulation*. In this net the signals propagate along interconnections in the form of quasi-monochromatic pulses at q different frequencies. There are arguments in favor of this idea. First of all, the frequency-phase modulation is more convenient for optical signal processing. It allows one to back down an artificial adaptation of an optical network to amplitude modulated signals. Second, when signals with q different frequencies can propagate along one interconnection this is an analog of the *channel multiplexing*. In fact, this allows one to reduce the number of interconnections by a factor of q^2 . Note that interconnections occupy nearly 98% of the area of neuro-chips.

In the center of this model the parametric 4-wave mixing process (FWM) is situated, which is well-known in *nonlinear optics* [Blo66]. However, in order this model to have good characteristics, an important condition needs to be added that should facilitate the propagation of a useful signal, and at the same time, suppress an internal noise. This condition is *the principle of incommensurability of frequencies* proposed in [KM02, FK01] in nonlinear optics terms.

The signal-noise analysis of this model, performed using the Chebyshev-Chernov statistical method [Che52, KMK00], showed that the storage capacity of the network was approximately q^2 times as much as the HM storage capacity. This network was called the *parametric neural network* (PNN).

The authors of [KLF02, MKL, KLM04] proposed the general *vector formalism* – universal description of PNN, not related directly to the optical model.

This formalism proved to be useful also for clear description of PGNN, although initially it was formulated in another terms. In this way one can easily establish relations between PGNN and PNN and also clarify the mechanisms responsible for outstanding recognizing properties of both models. The reason is the local architecture of both networks, which suppresses system's internal noise. In other q -valued models there is no such suppression.

Potts–Glass ANN

Following [KLF02, MKL, KLM04], we describe PGNN in terms of the vector formalism and compare it with PNN. PGNN consists of N neurons, each of which can be at q different states. In order to describe the q different states of neurons we can use the set of q D vectors of a special type, termed *Potts vectors*. The l th state of a neuron is described by a column-vector $\mathbf{d}_l \in \mathbb{R}^q$,

$$\mathbf{d}_l = \frac{1}{q} \begin{pmatrix} -1 \\ \vdots \\ q-1 \\ \vdots \\ -1 \end{pmatrix}, \quad (l = 1, \dots, q).$$

The state of the i th neuron is described by a vector $\mathbf{x}_i = \mathbf{d}_{l_i}$, ($1 \leq l_i \leq q$). The state of the network as a whole X is determined by a set of N column-vectors $\mathbf{x}_i$: $X = (\mathbf{x}_1, \dots, \mathbf{x}_N)$. The p stored patterns are

$$X^{(\mu)} = (\mathbf{x}_1^{(\mu)}, \dots, \mathbf{x}_N^{(\mu)}), \quad \mathbf{x}_i^{(\mu)} = \mathbf{d}_{l_i^{(\mu)}}, \\ (1 \leq l_i^{(\mu)} \leq q, \quad \mu = 1, 2, \dots, p).$$

Since neurons are vectors, the local field $\mathbf{h}_i$ affecting the i th neuron is a vector too,

$$\mathbf{h}_i = \frac{1}{N} \mathbf{T}_{ij} \mathbf{x}_j.$$

The $(q \times q)$ -matrices $\mathbf{T}_{ij}$ describe the interconnections between the i th and the j th neurons. By analogy with the Hopfield model these matrices are chosen in a generalized Hebbian form:

$$\mathbf{T}_{ij} = (1 - \delta_{ij}) \mathbf{x}_i^{(\mu)} \mathbf{x}_j^{(\mu)+}, \quad (i, j = 1, \dots, N), \quad (3.35)$$

where $\mathbf{x}^+$ is q D row-vector and δ_{ij} is the Kronecker symbol. The matrix $\mathbf{T}_{ij}$ affects the vector $\mathbf{x}_j \in \mathbb{R}^q$, converting it into a linear combination of column-vectors $\mathbf{d}_l$. After summation over all j we get the local field $\mathbf{h}_i$ as a linear combination of vectors $\mathbf{d}_l$

$$\mathbf{h}_i = A_l^{(i)} \mathbf{d}_l.$$

Let k be the index related to the maximal coefficient: $A_k^{(i)} > A_l^{(i)}$ for all l . Then, by definition, the i th neuron at the next time step, $t + 1$, is oriented along a direction closest to the local field $\mathbf{h}_i$ at the time t :

$$\mathbf{x}_i(t + 1) = \mathbf{d}_k. \quad (3.36)$$

The system evolution consists of consequent changes of orientations of vector-neurons according to the rule (3.36). Following [KLF02, MKL, KLM04], we make the convention that if some of the coefficients $A_l^{(i)}$ are maximal simultaneously, and the neuron is in one of these *un-improvable* states, its state does not change. Then it is easy to show that during the evolution of the network its *Hamiltonian energy function*,

$$H(t) = -1/2(\mathbf{h}_i(t)\mathbf{x}_i(t)),$$

decreases. At the end the system reaches a local energy minimum. In this state all the neurons $\mathbf{x}_i$ are oriented in an un-improvable manner, and the evolution of the system come to its end. These states are the fixed-points of the system. The necessary and sufficient conditions for a configuration X to be a fixed-point is given by the following set of inequalities:

$$(\mathbf{x}_i\mathbf{h}_i) \geq (\mathbf{d}_l\mathbf{h}_i), \quad (\text{for all } l = 1, \dots, q; \ i = 1, \dots, N). \quad (3.37)$$

When $q = 2$, PGNN becomes the standard Hopfield model.

Storage Capacity of PGNN

Let us consider randomized patterns $\{X^{(\mu)}\}_1^p$. Suppose that the network starts from a distorted m th pattern [KLM04]

$$\tilde{X}^{(m)} = (\hat{b}_1\mathbf{x}_1^{(m)}, \hat{b}_2\mathbf{x}_2^{(m)}, \dots, \hat{b}_N\mathbf{x}_N^{(m)}).$$

The noise operator $\hat{b}_j$, with the probability b , changes the state of the vector $\mathbf{x}_j^{(m)}$, and with the probability $1 - b$ this vector remains unchanged. In other words, b is the probability of an error in a state of a neuron. The noise operators $\hat{b}_j$ are also independent.

The network recognizes the reference pattern X^m correctly if the output of the i th neuron defined by (3.36) is equal to $\mathbf{x}_i^{(m)}$. Otherwise, PGNN fails to recognize the pattern X^m . Let us estimate the probability of error in the recognition of m th pattern. Simple calculations show, that probability of inequality validity $(\mathbf{x}_i^{(m)}\mathbf{h}_i) < (\mathbf{d}_l\mathbf{h}_i)$ at $\mathbf{d}_l \neq \mathbf{x}_i^{(m)}$ can be expressed as

$$\text{Prob} \{ \xi < \eta \} = \text{Prob} \left\{ \frac{1}{N} \sum_{j \neq i}^N \xi_j < \frac{1}{N} \sum_{j \neq i}^N \sum_{\mu \neq m}^p \eta_j^{(\mu)} \right\}, \quad \text{where} \quad (3.38)$$

$$\eta_j^{(\mu)} = (\mathbf{d}_l - \mathbf{x}_i^{(m)}, \mathbf{x}_i^{(\mu)})(\mathbf{x}_j^{(\mu)}\hat{b}_j\mathbf{x}_j^{(m)}), \quad \xi_j = (\mathbf{x}_j^{(m)}\hat{b}_j\mathbf{x}_j^{(m)}).$$

The quantity ξ is the *useful signal*. It is related to the influence of the m th pattern onto the i th neuron. The partial random variables ξ_j are independent and identically distributed. The quantity η symbolizes the *inner noise*, related to distorting influence of all other patterns. Partial noise components $\eta_j^{(\mu)}$ are independent and identically distributed. It is easy to get the distributions for ξ_j and $\eta_j^{(\mu)}$:

$$\begin{cases} \xi_j = \begin{cases} (q-1)/q, 1-b \\ -1/q, b \end{cases}, \\ \eta_j^{(\mu)} = \begin{cases} (q-1)/q, 1/q^2 \\ 1/q, (q-1)/q^2 \\ 0, (q-2)/q \\ -1/q, (q-1)/q^2 \\ -(q-1)/q, 1/q^2 \end{cases}. \end{cases} \quad (3.39)$$

Consider the fact that at $q \gg 1$ the noise component $\eta_j^{(\mu)}$ is localized mainly at zero:

$$\text{Prob} \{ \eta_j^{(\mu)} = 0 \} = (q-2)/q \sim 1.$$

Total random variables ξ, η are asymptotic normally-distributed with the following parameters:

$$\begin{aligned} E(\xi) &= \frac{q-1}{q} - b, & E(\eta) &= 0, \\ D(\xi) &\rightarrow 0; & D(\eta) &= \frac{2(q-1)}{q^3} \cdot \alpha, \end{aligned} \quad (3.40)$$

where (as usual) the *loading parameter*, $\alpha = p/N$. Now the probability of recognition error of coordinate $\mathbf{x}_i^{(m)}$ can be calculated by integration of the area under the ‘tail’ of normally distributed η , where $\eta > E(\xi)$.

Here we can explain, why the storage capacity of PGNN is much larger than HM. The same considerations are valid for HM (see, e.g., [HKP93]). Again we get a useful signal ξ and an internal noise η and (3.38) for the probability of recognition-failure. These random quantities will asymptotic normal as sums of independent, identically distributed partial random components ξ_j and $\eta_j^{(\mu)}$. The distributions of these last components can be obtained from (3.39) at $q = 2$ (as PGNN transforms into HM in this case). Mean values and dispersions for ξ and η can be obtained from (3.40) in the same way. As a result we get for HM:

$$\begin{aligned} \xi_j &= \begin{cases} 1/2, 1-b \\ -1/2, b \end{cases}, & \eta_j^{(\mu)} &= \begin{cases} 1/2, 1/2 \\ -1/2, 1/2 \end{cases}, \\ E(\xi) &= \frac{1}{2} - b, & E(\eta) &= 0, \\ D(\xi) &\rightarrow 0; & D(\eta) &= \frac{\alpha}{4}. \end{aligned} \quad (3.41)$$

Comparing (3.41) with (3.39) and (3.40) demonstrates that the dispersion of internal noise for PGNN is much smaller than that for HM:

$$D_{PGNN}(\eta)/D_{HM}(\eta) = \frac{8(q-1)}{q^3} \ll 1, \text{ when } q \gg 1.$$

Already at $q \sim 10$ the internal noise dispersion for PGNN is an order of magnitude smaller than that for HM. Moreover, at $q \sim 10^2$ we have the fall of the dispersion four orders of magnitude! This defines PGNN superiority over HM.

Switching from one vector-coordinate situation to that with the whole pattern and using the standard approximation ([KL03],[MKL]) we get the expression for the probability of the error in the recognition of the pattern $X^{(m)}$,

$$\text{Pr}_{err} \sim \sqrt{Np} \exp \left(-\frac{N}{2p} \frac{q(q-1)}{2} (1-\bar{b})^2 \right), \quad \bar{b} = \frac{q}{q-1} b. \quad (3.42)$$

The expression sets the upper limit for the probability of recognition failure for PGNN. Then the asymptotically possible value of the storage capacity of PGNN is

$$p_c = \frac{N}{2 \ln N} \frac{q(q-1)}{2} (1-\bar{b})^2. \quad (3.43)$$

When $q = 2$, these expressions give the known estimates for HM. For $q > 2$ the storage capacity of PGNN is $q(q-1)/2$ times as large as the storage capacity of HM. In [Kan88] the same factor was obtained by fitting the results of numerical calculations.

Parametric ANN

Here we describe the AM-model both in nonlinear optics and in vector-formalism terms. We also set out the obtained results for this model [KLM04]. In the network the signals propagate along interconnections in the form of quasi-monochromatic pulses at q different frequencies

$$\{\omega_l\}_1^q \equiv \{\omega_1, \omega_2, \dots, \omega_q\}. \quad (3.44)$$

The model is based on a parametric neuron that is a cubic nonlinear element capable to transform and generate frequencies in the parametric FWM-processes

$$\omega_i - \omega_j + \omega_k \longrightarrow \omega_r.$$

Schematically, this model of a neuron can be assumed as a device that is composed of a summator of input signals, a set of q ideal frequency filters $\{\omega_l\}_1^q$, a block comparing the amplitudes of the signals and q generators of quasi-monochromatic signals $\{\omega_l\}_1^q$.

Let $\{K^{(\mu)}\}_1^p$ be a set of patterns each of which is a set of quasi-monochromatic pulses with frequencies defined by (3.44) and amplitudes equal to ± 1 :

$$K^{(\mu)} = (\kappa_1^{(\mu)}, \dots, \kappa_N^{(\mu)}), \quad \kappa_i^{(\mu)} = \pm \exp(i\omega_{l_i^{(\mu)}} t), \quad (3.45)$$

$$(\mu = 1, \dots, p; \quad i = 1, \dots, N; \quad 1 \leq l_i^{(\mu)} \leq q).$$

The memory of the network is localized at interconnections T_{ij} , $i, j = 1, \dots, N$, which accumulate the information about the states of i th and j th neurons in all the p patterns. We suppose that the interconnections are dynamic and that they are organized according to the *Hebbian rule*:

$$T_{ij} = (1 - \delta_{ij})\kappa_i^{(\mu)}\kappa_j^{(\mu)*}, \quad (i, j = 1, \dots, N). \quad (3.46)$$

The network operates as follows. A quasi-monochromatic pulse with a frequency ω_{l_j} that is propagating along the (ij) th interconnection from the j th neuron to the i th one, takes part in FWM-processes with the pulses stored at the interconnection,

$$\omega_{l_i^{(\mu)}} - \omega_{l_j^{(\mu)}} + \omega_{l_j} \longrightarrow \{\omega_l\}_1^q.$$

The amplitudes ± 1 need to be multiplied. Summing up the results of these partial transformations over all patterns, $\mu = 1, \dots, p$, we get a packet of quasi-monochromatic pulses, where all the frequencies from the set (3.44) are present. This packet is the result of transformation of the pulse ω_{l_j} by the interconnection T_{ij} , and it comes to the i th neuron. All such packets are summarized in this neuron. The summarized signal propagates through q parallel ideal frequency filters. The output signals from the filters are compared with respect to their amplitudes. The signal with the maximal amplitude activates the i th neuron ('winner-take-all'). As a result, it generates an output signal whose frequency and phase are the same as the frequency and the phase of the activating signal.

Generally, when three pulses interact, under a FWM-process always the fourth pulse appears. The frequency of this pulse is defined by the conservation laws only. However, in order that the above mentioned model works as an AM, an important condition needs to be added, which has to facilitate the propagation of the useful signal, and at the same time, to suppress external noise. This condition is the *principle of incommensurability of frequencies* proposed in [KM02, FK01]: *no combinations $\omega_l - \omega_{l'} + \omega_{l''}$ can belong to the set (3.44), when all the frequencies are different.*

This network is called *the parametric neural network* (PNN). Generally speaking, there are different parametric FWM-processes complying with the principle of incommensurability of frequencies. However, better results can be obtained for the parametric FWM-process

$$\omega_l - \omega_{l'} + \omega_{l''} = \begin{cases} \omega_l, & \text{when } l' = l''; \\ \rightarrow 0, & \text{otherwise.} \end{cases} \quad (3.47)$$

This architecture is called PNN-2 (another architecture, PNN-1, was examined in [KM02, FK01]). Here we investigate the abilities of PNN-2.

Vector Formalism for PNN-2

In order to describe the q different states (3.44) of neurons we use the set of basis vectors $\mathbf{e}_l$ in the space R^q , $q \geq 1$,

$$\mathbf{e}_l = \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix}, \quad (l = 1, \dots, q).$$

The state of the i th neuron is described by a vector $\mathbf{x}_i$ [KLM04],

$$\mathbf{x}_i = x_i \mathbf{e}_{l_i}, \quad x_i = \pm 1, \quad \mathbf{e}_{l_i} \in \mathbb{R}^q, \quad \begin{cases} 1 \leq l_i \leq q; \\ i = 1, \dots, N. \end{cases} \quad (3.48)$$

The factor x_i denotes the signal phase. The state of the network as a whole X is determined by a set of N q D vectors $\mathbf{x}_i$: $X = (\mathbf{x}_1, \dots, \mathbf{x}_N)$. The p stored patterns are

$$\begin{aligned} X^{(\mu)} &= (\mathbf{x}_1^{(\mu)}, \mathbf{x}_2^{(\mu)}, \dots, \mathbf{x}_N^{(\mu)}), \quad \mathbf{x}_i^{(\mu)} = x_i^{(\mu)} \mathbf{e}_{l_i^{(\mu)}}, \\ (x_i^{(\mu)} &= \pm 1, \quad 1 \leq l_i^{(\mu)} \leq q, \quad \mu = 1, \dots, p), \end{aligned}$$

and the local field is

$$\mathbf{h}_i = \frac{1}{N} \mathbf{T}_{ij} \mathbf{x}_j. \quad (3.49)$$

The $(q \times q)$ -matrix $\mathbf{T}_{ij}$ describes the interconnection between the i th and the j th neurons. This matrix affects the vector $\mathbf{x}_j \in \mathbb{R}^q$, converting it into a linear combination of basis vectors $\mathbf{e}_l$. This combination is an analog of the packet of quasi-monochromatic pulses that come from the j th neuron to the i th one after transformation into the interconnection. To satisfy the conditions (3.46) and (3.47), we need to take the matrices $\mathbf{T}_{ij}$ as

$$\mathbf{T}_{ij} = (1 - \delta_{ij}) \mathbf{x}_i^{(\mu)} \mathbf{x}_j^{(\mu)+}, \quad (i, j = 1, \dots, N). \quad (3.50)$$

Note, that the structure of this expression is similar to that of (3.120).

The dynamic rule is left as earlier: the i th neuron at the time $t + 1$ is oriented along a direction mostly close to the local field $\mathbf{h}_i(t)$. However the expressions will differ from (3.36). Indeed, with the aid of (3.50), we write $\mathbf{h}_i$ in the form more convenient for analysis:

$$\mathbf{h}_i(t) = A_l^{(i)} \mathbf{e}_l, \quad A_l^{(i)} \sim (\mathbf{e}_l \mathbf{x}_i^{(\mu)}) (\mathbf{x}_j^{(\mu)} \mathbf{x}_j(t)). \quad (3.51)$$

Let k be the index relating to the amplitude that is maximal *in modulus* in the series (3.51): $|A_k^{(i)}| > |A_l^{(i)}|$ for all l . Then according to the above definition,

$$\mathbf{x}_i(t + 1) = \text{sgn}(A_k^{(i)}) \mathbf{e}_k. \quad (3.52)$$

The evolution of the system consists of consequent changes of orientations of vector-neurons according to the rule (3.52). The necessary and sufficient conditions for a configuration X to be a fixed-point is fulfillment of the set of inequalities (compare with (3.121)):

$$(\mathbf{x}_i \mathbf{h}_i) \geq |(\mathbf{e}_l \mathbf{h}_i)|, \quad (\text{for all } l = 1, \dots, q; \quad i = 1, \dots, N).$$

Storage Capacity of PNN-2

All these considerations are identical to those for PGNN. Differences appear only because of neurons are defined now not only by vectors, but also by scalars ± 1 . The distorted m th pattern has the form

$$\tilde{X}^{(m)} = (a_1 \hat{b}_1 \mathbf{x}_1^{(m)}, a_2 \hat{b}_2 \mathbf{x}_2^{(m)}, \dots, a_N \hat{b}_N \mathbf{x}_N^{(m)}).$$

Here $\{a_i\}_1^N$ and $\{\hat{b}_i\}_1^N$ define a *phase noise* and a *frequency noise* respectively: a_i is a random value that is equal to -1 or $+1$ with the probabilities a and $1 - a$ respectively; b is the probability that the operator $\hat{b}_i$ changes the state of the vector $\mathbf{x}_i^{(m)} = x_i^{(m)} \mathbf{e}_{l_i^{(m)}}$, and $1 - b$ is the probability that this vector remains unchanged.

The amplitudes $A_l^{(i)}$ (3.51) have the form [KLM04]

$$A_l^{(i)} \sim \begin{cases} x_i^{(m)} \sum_{j \neq i}^N \xi_j + \sum_{j \neq i}^N \sum_{\mu \neq m}^p \eta_j^{(\mu)}, & l = l_i^{(m)}, \\ \sum_{j \neq i}^N \sum_{\mu \neq m}^p \eta_j^{(\mu)}, & l \neq l_i^{(m)}, \end{cases}$$

where

$$\eta_j^{(\mu)} = a_j (\mathbf{e}_l \mathbf{x}_i^{(\mu)}) (\mathbf{x}_j^{(\mu)} \hat{b}_j \mathbf{x}_j^{(m)}), \quad \xi_j = a_j (\mathbf{x}_j^{(m)} \hat{b}_j \mathbf{x}_j^{(m)}), \\ (j(\neq i) = 1, \dots, N, \quad \mu(\neq m) = 1, \dots, p).$$

When the patterns $\{X^{(\mu)}\}_1^p$ are uncorrelated, the quantities ξ_j and $\eta_j^{(\mu)}$ are independent random variables described by the probability distributions (compare with (3.39))

$$\xi_j = \begin{cases} +1, & (1-b)(1-a) \\ 0, & b \\ -1 & (1-b)a \end{cases}, \quad \eta_j^{(\mu)} = \begin{cases} +1, & 1/2q^2 \\ 0, & 1 - 1/q^2 \\ -1 & 1/2q^2 \end{cases}.$$

As in the case of PGNN, when $q \gg 1$ the noise component $\eta_j^{(\mu)}$ is localized mainly at zero:

$$\text{Prob} \left\{ \eta_j^{(\mu)} = 0 \right\} = 1 - 1/q^2 \sim 1.$$

Equation (3.40) will now transform into:

$$\begin{aligned} E(\xi) &= (1 - 2a)(1 - b), & E(\eta) &= 0, \\ D(\xi) &\rightarrow 0, & D(\eta) &= \frac{1}{q^2} \cdot \alpha. \end{aligned}$$

When $q \gg 1$, the dispersion of internal noise for PNN-2 is even smaller than for PGNN:

$$D_{PNN}(\eta)/D_{PGNN}(\eta) = 1/2, \text{ when } q \gg 1.$$

In the long run this determines the superiority of PNN-2 over PGNN in memory capacity and noise immunity. It is convenient here to mention mechanisms

that suppress internal noises. They are identical in both models, but we will demonstrate them on the PNN example.

When signal propagates it interacts with frequencies, stored in interconnection

$$\omega_{l_i^{(\mu)}} - \omega_{l_j^{(\mu)}} + \omega_{l_j} \longrightarrow \{\omega_l\}_1^q.$$

In addition, the principal of frequencies incommensurability (3.47) should be fulfilled. It can be formulated in vector terms as:

$$\mathbf{x}_i^{(\mu)} \mathbf{x}_j^{(\mu)+} \mathbf{x}_j = \begin{cases} \mathbf{x}_i^{(\mu)}, & \text{when } l_j^{(\mu)} = l_j; \\ 0, & \text{otherwise.} \end{cases}$$

One can see from the last equation, that the largest part of propagated signals will be suppressed. It happens because the interconnection chooses the only one combinations of indices $l_j^{(\mu)}$ and l_j from all possible ones, where indices coincide (other combinations give zero). In other words, the interconnection filters signals. It is the main reason why the largest part of internal noise η is localized at zero.

The similar filtration happens also in PGNN. The difference is that in PGNN the signal always propagates through the interconnection. But when indices $l_j^{(\mu)}$ and l_j coincide, the signal is attributed with large positive amplitude ~ 1 . If indices do not coincide, the signal is attributed with small negative amplitude $\sim -1/q$. This signal filtration leads to suppression of internal noise in PGNN. In all another q -valued models of associative memory this filtration is absent.

At the end of consideration of PNN-2 we give the expressions for noise immunity and storage capacity similar to (3.42) and (3.43):

$$\text{Pr}_{err} \sim \sqrt{Np} \exp \left(-\frac{N(1-2a)^2}{2p} \cdot q^2(1-b)^2 \right), \quad (3.53)$$

$$p_c = \frac{N(1-2a)^2}{2 \ln N} \cdot q^2(1-b)^2. \quad (3.54)$$

When $q = 1$, (3.53)–(3.54) transform into well-known results for the standard Hopfield model (in this case there is no frequency noise, $b = 0$). When q increases, the probability of the error (3.53) decreases exponentially, i.e., the noise immunity of PNN increases noticeably. At the same time the storage capacity of the network increases proportionally to q^2 . In contrast to the Hopfield model the number of the patterns p can be much greater than the number of neurons.

For example, let us set a constant value $\text{Pr}_{err} = 0.01$. In the Hopfield model with this probability of the error we can recognize any of $p = N/10$ patterns, each of which is less then 30% noisy. At the same time, PNN-2 with $q = 64$ allows us to recognize any of $p = 5N$ patterns with 90% noise, or any of $p = 50N$ patterns with 65% noise. Computer simulations done by [KLM04] confirm these results.

The memory capacity in PNN-2 is twice as large as that in PGNN. Evidently, it is connected with the fact, that for the same q the number of different states of neurons in PNN-2 is twice as large as that in PGNN. In general, both models have very similar characteristics.

3.1.5 Small-World AM-Neural Nets

Recall that ANNs have been used as a model for associative memory since the 80's, and a considerable amount of work has been made in the field (see, e.g., [Ami89, Per92]). Most of this work regards both the simulation and the theory of completely connected networks, as well as networks with a random dilution of the connectivity. It is known that particular prescriptions for the determination of the synaptic weights enable these systems to successfully retrieve a pattern out of a set of memorized ones. This behavior is observed in the system up to a certain value of the number of stored patterns, beyond which the network becomes unable to retrieve any of them. For reasons of simplicity of the models and their analytical tractability, complex architectures of the networks, more akin to those found in biological neural systems, have been largely left out of the theoretical analysis. Fortunately, since a few years ago, a class of models that has come to be known as 'complex networks' began to be thoroughly studied. Complex networks seem more compatible with the geometrical properties of many biological and social phenomena than regular lattices, random networks, or completely connected systems [WS98, BA99, New00, Wat99]. Already in the seminal work of Watts and Strogatz [WS98], whose small-world model combines properties of regular and random networks, it was observed that the neural system of the nematode *C. elegans* shares topological properties with this model networks.

Following [MAK04], in this subsection we study a neural network built upon the *Watts-Strogatz model* for small worlds. The model interpolates between regular and random networks by means of a parameter p , which characterizes the disorder of the network. The construction, as formulated in [WS98], begins with a 1D regular lattice of N nodes, each one linked to its K nearest neighbors to the right and to the left, and with periodic boundary conditions. With probability p , each one of the right-pointing links, of every node, is rewired to a randomly chosen node in the network. Self connections and repeated connections are not allowed. The result is a disordered network, defined by the set N, K, p , that lies between a regular lattice ($p = 0$) and a random graph ($p = 1$). A wide range of these networks displays high local clusterization and short average distance between nodes, as many real complex networks. They can be defined by the *connectivity matrix* c_{ij} , where $c_{ij} = 1$ if there is a link between nodes i and j , and $c_{ij} = 0$ otherwise. We use this matrix to establish the synaptic connections between neurons, at variance from the traditional Hopfield model, where the network is completely connected and the connectivity matrix is $c_{ij} = 1$, for all i, j . At $p = 1$ it coincides with the standard diluted disordered networks, that have also been considered in

the literature, in which randomly chosen elements in the connectivity matrix are set to zero.

The biological neuron carries out an operation on the inputs provided by other neurons, and it produces an output. A transformation of this continuous output into a binary variable makes it possible to formulate a simplified model in which the neurons are logical elements [Ami89]. In this binary representation, the state of each neuron is characterized by a single variable s_i . This variable can take two values representing the active and the inactive states,

$$s_i = \begin{cases} 1, & \text{if the neuron is active,} \\ -1, & \text{if the neuron is inactive.} \end{cases}$$

The purpose of an AM-model, is to retrieve some patterns that have been stored in the network by an unspecified learning process. The stored—or *memorized*—patterns are represented by network states ξ^μ , where $\mu = 1, \dots, M$ labels the different patterns and M is their number. As usual, the patterns are generated at random, assigning with equal probability $1/2$ the values $\xi_i^\mu = \pm 1$. The patterns are uncorrelated and thus orthogonal in large networks [MAK04]

$$\frac{1}{N} \xi_i^\mu \xi_i^\nu = \delta_{\mu\nu}.$$

The state of the neurons is updated asynchronously, as in *Glauber dynamics*. At each simulation step, a neuron is chosen at random, and its new state is determined by the local field,

$$\begin{aligned} h_i &= \omega_{ij} s_j, & \text{according to} \\ s_i &= \text{sign}(h_i). \end{aligned}$$

The synaptic weights ω_{ij} of the connections are given by the *Hebbian rule*, restricted to the synapsis actually present in the network, as given by the connectivity matrix,

$$w_{ij} = \frac{1}{N} c_{ij} \xi_i^\mu \xi_j^\mu, \quad (i, j = 1, \dots, N). \quad (3.55)$$

Note that as the network model does not allow self connections the diagonal matrix elements are null. By definition, the synaptic matrix is symmetric.

The above dynamics is deterministic, and the network is not subject to thermal fluctuations. We will only consider the effects of a small amount of additive noise to verify the robustness of our results. A full discussion of the effect of a finite temperature in the dynamics will be left for future work. The stochastic asynchronous update, though, prevents the system from having limit cycles, and the only attractors are fixed-points. The stored patterns ξ^μ are, by construction of the synaptic weights (3.55), fixed-points of the dynamics due to the orthogonality condition. In the model, ‘memory’ is the capacity of the network to retrieve one of the stored patterns from an arbitrary

initial condition. As in traditional models, the reversed patterns $(-\xi_i)$, as well as a wealth of symmetric and asymmetric mixtures of patterns, are also equilibria of the system and play a significant role in its behavior as a memory device.

Disordered Topology Effect

We have performed extensive numerical simulations of the system, starting from a random unbiased initial condition. After a transient, a fixed-point is reached, whence no further changes occur to any neuron. In order to measure the efficacy of the network to recall a number M of stored random patterns, we define an *efficacy* φ as the fraction of realizations in which one of the stored patterns is retrieved. We get the surprising non-monotonic behavior with the disorder parameter p , which has been observed before in a problem of biased diffusion [Zan02a], and in an *Ising-spin model* [SLR02], both with asymmetric interactions.

The fact that the transition between the remembering and the non-remembering phases occurs at a finite value of the disorder parameter is very interesting, since a few dynamical systems based on small-world architectures show it [KA01, Zan02b]. This occurs in spite of the fact that the average distance between nodes, the main geometrical property of the Watts–Strogatz model, has a transition at $p = 0$ [BW00]. Indeed, for several Ising-like systems, which bear some similarities with artificial neural networks, a phase transition occurs at $p = 0$ [BW00, ZZ03, Her02, KHH01].

In order to understand the finite size effects in the system, and the behavior of the transition in the limit of an infinite system, we have made simulations on systems of different sizes. We have chosen to keep the connectivity parameter of the model constant through all the results we show, $K = 100$. In this regard, our results correspond to a neural network characterized by certain properties at the local level, for example the average connectivity of each neuron ($2K$ in our systems). Our finite size analysis shows the behavior of these networks in systems of increasing size N and in the limit $N \rightarrow \infty$.

Based on numerical evidence, we find that the dependence of the efficacy on the system size can be built into a scaling function [MAK04]

$$\varphi(p, N) = \Phi[(p - p_c)N^\alpha].$$

At the point of crossing of the curves, φ becomes independent of N .

Since the order parameter is not singular at the transition, we can expand Φ as a Taylor series around the critical control parameter p_c :

$$\varphi(p, N) = \Phi(0) + \Phi'(0)(p - p_c)N^\alpha,$$

to first order in $(p - p_c)$. Defining $\tilde{\varphi} = \varphi - \varphi(p_c)$ and $\tilde{p} = p - p_c$ we can write:

$$\left. \frac{\partial \tilde{\varphi}}{\partial \tilde{p}}(N) \right|_{\tilde{p}=0} = \Phi'(0)N^\alpha.$$

Except in the relatively narrow range of p where $\varphi \approx 1$, the system fails to retrieve any stored pattern in a significant fraction of the realizations: almost always when the network is very ordered (down to $p = 0$), and about 12% of the times when the network is very disordered (up to $p = 1$). What happens in the phase space as the network architecture changes? What happens to the trajectories, and why are the patterns missed? It seems natural to expect that the energy landscape is different for $p = 0$ than for $p = 1$. To address this problem we turn our attention to the properties of the overlaps of the equilibrium state with the memorized patterns. Suppose that after a transient the network has reached a fixed-point ζ . We define the overlap of this fixed-point with the patterns as

$$\theta^\mu = \frac{1}{N} |\xi_i^\mu \zeta_i|.$$

Note that if the fixed-point is a stored pattern, $\zeta = \xi^\nu$, then $\theta^\nu = 1$. In order to determine the type of fixed-points that are reached when the network misses the patterns, we measure the overlap θ^μ of the fixed-point with the stored patterns ξ^μ . The probability distribution $P(\theta)$ of these overlaps gives information on the kind of mixture that the fixed-point is. The fixed-points of these systems consist of very asymmetric mixtures.

The previous analysis unveiled the structure of the phase space and the difference between the low and the high p regimes. Still, what is the reason for the catastrophic loss of memory below the critical value of disorder? We have found that, for low values of disorder, the fixed-points retrieve scattered pieces of several stored patterns. These fixed-points consist of localized regions that overlap with different patterns. Indeed, at $p = 0$, the network is topologically very clusterized, and there exist local neighborhoods relatively isolated from each other. These neighborhoods begin to disappear by the action of the shortcuts provided by the random rewiring at higher values of p , until the whole system becomes essentially a single neighborhood. Then, at $p = 0$, from an arbitrary initial condition, different regions of the network eventually align themselves with different patterns. The final result is a completely asymmetric mixture, impossible to classify due to the arbitrariness of its origin and nature. The existence of asymmetric mixtures as attractors in this kind of AM-model have been observed before (see, e.g., [Ami89]). But since they are very rare in the completely random or in the completely connected networks, they are very difficult to observe. In the present context, however, they play an essential role in the destruction of the ability of the system to retrieve the patterns.

In order to quantify this, we proceed to define a correlation measure that provides a clear picture of the situation. We introduce the difference of the fixed-point ζ with a given pattern:

$$d_i^\mu = \xi_i^\mu \zeta_i = \begin{cases} 1 & \text{if } \xi_i^\mu = \zeta_i, \\ -1 & \text{if } \xi_i^\mu \neq \zeta_i. \end{cases}$$

Then we define a local magnetization for the difference vector d^μ , for every node i :

$$m_i^\mu = \frac{1}{1 + k_i} \left| d_i^\mu + \sum_{j \in i} d_j^\mu \right|,$$

where k_i is the set of neighbors of node i . The local magnetization m_i^μ measures the local alignment with the μ pattern or its reversed companion. The maximum value $m_i^\mu = 1$ arises when $d_j^\mu = d_i^\mu$ for all $j \in k_i$. The presence of connected domains where the fixed-point ζ overlaps with the ξ^μ pattern should be detected as short range correlations between the local magnetizations. The correlation between the local magnetizations of the difference vector with the μ pattern are then defined as [MAK04]

$$C^\mu = \frac{1}{N} \sum_{i=1}^N \frac{1}{k_i} \sum_{j \in i} m_i^\mu m_j^\mu.$$

To be able to capture the existence of correlations in the difference with patterns that appear in the mixture that makes up the fixed-point ζ , we have the *maximum correlation*

$$C = \max_{\mu} \{C^\mu\}.$$

3.2 Local Neurodynamics

3.2.1 Microscopic Neurodynamics and Quantum Brain

Biochemistry of Microtubules

Recent developments/efforts to understand aspects of the brain function at the *sub-neural* level are discussed in [Nan95]. Microtubules (MTs), protein polymers constructing the cytoskeleton of a neuron, participate in a wide variety of dynamical processes in the cell. Of special interest for this subsection is the MTs participation in bio-information processes such as *learning* and *memory*, by possessing a well-known binary error-correcting code $[K_1(13, 2^6, 5)]$ with 64 words. In fact, MTs and DNA/RNA are *unique* cell structures that possess a code system. It seems that the MTs' code system is strongly related to a kind of *mental code* in the following sense. The MTs' periodic paracrystalline structure make them able to support a *superposition* of coherent quantum states, as it has been recently conjectured by Hameroff and Penrose [HP96], representing an *external* or *mental order*, for sufficient time needed for *efficient quantum computing*.

Living organisms are collective assemblies of cells which contain collective assemblies of organized material, including membranes, organelles, nuclei, and the *cytoplasm*, the bulk interior medium of living cells. Dynamic rearrangements of the cytoplasm within *eucaryotic cells*, the cells of all animals and almost all plants on Earth, account for their changing shape, movement, etc.

This extremely important cytoplasmic structural and dynamical organization is due to the presence of networks of interconnected protein polymers, which are referred to as the *cytoskeleton* due to their bone-like structure [HP96, Dus84]. The cytoskeleton consists of MT's, actin microfilaments, intermediate filaments and an *organizing complex*, the *centrosome* with its chief component the *centriole*, built from two bundles of microtubules in a separated **T** shape. Parallel-arrayed MTs are interconnected by cross-bridging proteins (*MT-Associated Proteins*: MAPs) to other MTs, organelle filaments and membranes to form *dynamic networks* [HP96, Dus84]. MAPs may be contractile, structural, or enzymatic. A very important role is played by contractile MAPs, like dynein and kinesin, through their participation in cell movements as well as in intra-neural, or axoplasmic transport which moves material and thus is of fundamental importance for the *maintenance* and *regulation* of *synapses* (see, e.g., [Ecc64]). The structural bridges formed by MAPs stabilize MTs and prevent their disassembly. The MT-MAP 'complexes' or *cytoskeletal networks* determine the cell architecture and dynamic functions, such a *mitosis*, or *cell division*, *growth*, *differentiation*, *movement*, and for us here the very crucial, *synapse formation and function*, all essential to the living state. It is usually said that *microtubules* are ubiquitous through the entire biology [HP96, Dus84].

MTs are hollow cylinders comprised of an exterior surface of cross-section diameter 25 nm ($1\text{ nm} = 10^{-9}$ meters) with 13 arrays (protofilaments) of protein dimers called tubulins [Dus84]. The interior of the cylinder, of cross-section diameter 14 nm , contains *ordered water* molecules, which implies the existence of an electric dipole moment and an electric field. The arrangement of the dimers is such that, if one ignores their size, they resemble triangular lattices on the MT surface. Each dimer consists of two hydrophobic protein pockets, and has an unpaired electron. There are two possible positions of the electron, called α and β *conformations*. When the electron is in the β -conformation there is a 29° distortion of the electric dipole moment as compared to the α conformation.

In standard models for the simulation of the MT dynamics [STZ93, SZT98], the 'physical' DOF – relevant for the description of the energy transfer – is the projection of the electric dipole moment on the longitudinal symmetry axis (x -axis) of the MT cylinder. The 29° distortion of the β -conformation leads to a displacement u_n along the x -axis, which is thus the relevant physical DOF.

There has been speculation for quite some time that MTs are involved in information processing: it has been shown that the particular geometrical arrangement (packing) of the tubulin protofilaments obeys an error-correcting mathematical code known as the $K_2(13, 2^6, 5)$ -code [KHS93]. Error correcting codes are also used in classical computers to protect against errors while in quantum computers special error correcting algorithms are used to protect against errors by preserving quantum coherence among qubits.

Information processing occurs via interactions among the MT proto-filament chains. The system may be considered as similar to a model of *interacting Ising chains* on a triangular lattice, the latter being defined on the plane stemming from filleting open and flattening the cylindrical surface of MT. Classically, the various dimers can occur in either α or β conformations. Each dimer is influenced by the neighboring dimers resulting in the possibility of a transition. This is the basis for classical information processing, which constitutes the picture of a (classical) cellular automaton.

Kink Soliton Model of MT-Dynamics

The *quantum nature* of an MT network results from the *assumption* that each dimer finds itself in a *superposition* of α and β conformations. Viewed as a *two-state quantum mechanical system*, the MT tubulin dimers couple to conformational changes with $10^{-9} - 10^{-11}$ sec transitions, corresponding to an angular frequency $\omega \sim \mathcal{O}(10^{10}) - \mathcal{O}(10^{12})$ Hz [Nan95].

The *quantum computer* character of the MT network [Pen89] results from the assumption that each dimer finds itself in a superposition of α and β conformations [Ham87]. There is a macroscopic coherent state among the various chains, which lasts for $\mathcal{O}(1 \text{ sec})$ and constitutes the ‘preconscious’ state [Nan95]. The interaction of the chains with (non-critical stringy) quantum gravity, then, induces self-collapse of the wave function of the coherent MT network, resulting in quantum computation.

In [EMN92, EMN99, MN95a, MN95b, Nan95] the authors assumed that the collapse occurs mainly due to the interaction of each chain with quantum gravity, the interaction from neighboring chains being taken into account by including mean-field interaction terms in the dynamics of the displacement field of each chain. This amounts to a modification of the effective potential by anharmonic oscillator terms. Thus, the effective system under study is 2D, possessing one space and one time coordinate.

Let u_n be the displacement field of the n th dimer in a MT chain. The continuous approximation proves sufficient for the study of phenomena associated with energy transfer in biological cells, and this implies that one can make the replacement

$$u_n \rightarrow u(x, t), \quad (3.56)$$

with x a spatial coordinate along the longitudinal symmetry axis of the MT. There is a time variable t due to fluctuations of the displacements $u(x)$ as a result of the dipole oscillations in the dimers.

The effects of the neighboring dimers (including neighboring chains) can be phenomenologically accounted for by an effective potential $V(u)$. In the kink-soliton model³ of ref. [STZ93, SZT98] a double-well potential was used,

³ Recall that kinks are solitary (non-dispersive) waves arising in various 1D (bio)physical systems.

leading to a classical kink solution for the $u(x, t)$ field. More complicated interactions are allowed in the picture of Ellis *et al.*, where more generic polynomial potentials have been considered.

The effects of the surrounding water molecules can be summarized by a *viscous force* term that damps out the dimer oscillations,

$$F = -\gamma \partial_t u, \quad (3.57)$$

with γ determined phenomenologically at this stage. This friction should be viewed as an environmental effect, which however does not lead to energy dissipation, as a result of the non-trivial solitonic structure of the ground-state and the non-zero constant force due to the electric field. This is a well known result, directly relevant to energy transfer in biological systems.

In mathematical terms the effective equation of motion for the relevant field DOF $u(x, t)$ reads:

$$u''(\xi) + \rho u'(\xi) = P(u), \quad (3.58)$$

where $\xi = x - vt$, $u'(\xi) = du/d\xi$, v is the velocity of the soliton, $\rho \propto \gamma$ [STZ93, SZT98], and $P(u)$ is a polynomial in u , of a certain degree, stemming from the variations of the potential $V(u)$ describing interactions among the MT chains. In the mathematical literature there has been a classification of solutions of equations of this form. For certain forms of the potential the solutions include *kink solitons* that may be responsible for dissipation-free energy transfer in biological cells:

$$u(x, t) \sim c_1 (\tanh[c_2(x - vt)] + c_3), \quad (3.59)$$

where c_1, c_2, c_3 are constants depending on the parameters of the dimer lattice model. For the form of the potential assumed in the model of [STZ93, SZT98] there are solitons of the form $u(x, t) = c'_1 + \frac{c'_2 - c'_1}{1 + e^{c'_3(c'_2 - c'_1)(x - vt)}}$, where again c'_i , $i = 1, \dots, 3$ are appropriate constants.

A *semiclassical quantization* of such solitonic states has been considered by Ellis *et al.*. The result of such a quantization yields a *modified soliton equation* for the (quantum corrected) field $u_q(x, t)$ [TF91]

$$\partial_t^2 u_q(x, t) - \partial_x^2 u_q(x, t) + \mathcal{M}^{(1)}[u_q(x, t)] = 0, \quad (3.60)$$

with the notation

$$M^{(n)} = e^{\frac{1}{2}(G(x, y, t) - G_0(x, y)) \frac{\partial^2}{\partial z^2}} U^{(n)}(z) \big|_{z=u_q(x, t)}, \quad U^{(n)} \equiv d^n U / dz^n.$$

The quantity U denotes the potential of the original soliton Hamiltonian, and $G(x, y, t)$ is a bilocal field that describes quantum corrections due to the modified boson field around the soliton. The quantities $M^{(n)}$ carry information about the quantum corrections. For the kink soliton (3.59) the quantum corrections (3.60) have been calculated explicitly in [TF91], thereby providing us with a concrete example of a large-scale quantum coherent state.

A typical propagation velocity of the kink solitons (e.g., in the model of [STZ93, SZT98]) is $v \sim 2$ m/sec, although, models with $v \sim 20$ m/sec have also been considered. This implies that, for moderately long microtubules of length $L \sim 10^{-6}$ m, such kinks transport energy without dissipation in

$$t_F \sim 5 \times 10^{-7} \text{ s.} \quad (3.61)$$

Such time scales are comparable to, or smaller in magnitude than, the decoherence time scale of the above-described coherent (solitonic) states $u_q(x, t)$. This implies the possibility that fundamental quantum mechanical phenomena may then be responsible for frictionless energy (and signal) transfer across microtubular arrangements in the cell [Nan95].

Open Liouville Neurodynamics and Self-Similarity

Recall that neurodynamics has its physical behavior both on the *macroscopic*, classical, *inter-neuronal level*, and on the *microscopic*, quantum, *intra-neuronal level*. On the *macroscopic* level, various models of neural networks (NNs, for short) have been proposed as goal-oriented models of the specific neural functions, like for instance, function-approximation, pattern-recognition, classification, or control (see, e.g., [Hay94]). In the physically-based, Hopfield-type models of NNs [Hop82, Hop84] the information is stored as a content-addressable memory in which synaptic strengths are modified after the Hebbian rule (see [Heb49]). Its retrieval is made when the network with the symmetric couplings works as the point-attractor with the fixed-points. Analysis of both *activation* and *learning dynamics* of Hopfield-Hebbian NNs using the techniques of statistical mechanics [DHS91], gives us with the most important information of storage capacity, role of noise and recall performance.

On the other hand, on the general *microscopic* intra-cellular level, energy transfer across the cells, without dissipation, had been first conjectured to occur in biological matter by [FK83]. The phenomenon conjectured by them was based on their 1D superconductivity model: in 1D electron systems with holes, the formation of *solitonic structures* due to electron-hole pairing results in the transfer of electric current without dissipation. In a similar manner, Frölich and Kremer conjectured that energy in biological matter could be transferred without dissipation, if appropriate solitonic structures are formed inside the cells. This idea has lead theorists to construct various models for the energy transfer across the cell, based on the formation of *kink* classical solutions (see [STZ93, SZT98]).

The interior of living cells is structurally and dynamically organized by *cytoskeletons*, i.e., networks of protein polymers. Of these structures, *microtubules* (MTs, for short) appear to be the most fundamental (see [Dus84]). Their dynamics has been studied by a number of authors in connection with the mechanism responsible for dissipation-free energy transfer. Hameroff and

Penrose [Ham87] have conjectured another fundamental role for the MTs, namely being responsible for *quantum computations* in the human neurons. [Pen89, Pen94, Pen97] further argued that the latter is associated with certain aspects of quantum theory that are believed to occur in the cytoskeleton MTs, in particular quantum superposition and subsequent collapse of the wave function of coherent MT networks. These ideas have been elaborated by [MN95a, MN95b] and [Nan95], based on the quantum-gravity EMN-language of [EMN92, EMN99] where MTs have been physically modeled as non-critical (SUSY) bosonic strings. It has been suggested that the neural MTs are the microsites for the emergence of stable, macroscopic quantum coherent states, identifiable with the *preconscious states*; stringy-quantum space-time effects trigger an organized collapse of the coherent states down to a specific or *conscious state*. More recently, [TVP99] have presented the evidence for biological self-organization and pattern formation during embryogenesis.

Now, we have two space-time biophysical scales of neurodynamics. Naturally the question arises: are these two scales somehow inter-related, is there a space-time self-similarity between them?

The purpose of this subsection is to prove the formal positive answer to the self-similarity question. We try to describe neurodynamics on both physical levels by the *unique form* of a single equation, namely *open Liouville equation*: NN-dynamics using its classical form, and MT-dynamics using its quantum form in the Heisenberg picture. If this formulation is consistent, that would prove the *existence* of the *formal neurobiological space-time self-similarity*.

Hamiltonian Framework

Suppose that on the macroscopic NN-level we have a conservative Hamiltonian system acting in a $2ND$ symplectic phase-space $T^*Q = \{q^i(t), p_i(t)\}$, ($i = 1 \dots N$) (which is the cotangent bundle of the NN-configuration manifold $Q = \{q^i\}$), with a Hamiltonian function $H = H(q^i, p_i, t) : T^*Q \times \mathbb{R} \rightarrow \mathbb{R}$. The conservative dynamics is defined by classical Hamiltonian canonical equations:

$$\dot{q}^i = \partial_p H, \quad \dot{p}_i = -\partial_q H. \quad (3.62)$$

Recall that within the conservative Hamiltonian framework, we can apply the formalism of classical Poisson brackets: for any two functions $A = A(q^i, p_i, t)$ and $B = B(q^i, p_i, t)$ their Poisson bracket is defined as

$$[A, B] = \left(\frac{\partial A}{\partial q^i} \frac{\partial B}{\partial p_i} - \frac{\partial A}{\partial p_i} \frac{\partial B}{\partial q^i} \right).$$

Conservative Classical System

Any function $A(q^i, p_i, t)$ is called a *constant* (or integral) of motion of the conservative system (3.62) if

$$\dot{A} \equiv \partial_t A + [A, H] = 0, \quad \text{which implies} \quad \partial_t A = -[A, H]. \quad (3.63)$$

For example, if $A = \rho(q^i, p_i, t)$ is a *density function* of ensemble phase-points (or, a probability density to see a state $x(t) = (q^i(t), p_i(t))$ of *ensemble* at a moment t), then equation

$$\partial_t \rho = -[\rho, H] = -iL \rho \quad (3.64)$$

represents the *Liouville theorem*, where L denotes the (Hermitian) *Liouville operator*

$$iL = [\dots, H] \equiv \left(\frac{\partial H}{\partial p_i} \frac{\partial}{\partial q^i} - \frac{\partial H}{\partial q^i} \frac{\partial}{\partial p_i} \right) = \text{div}(\rho \dot{\mathbf{x}}),$$

which shows that the conservative Liouville equation (3.64) is actually equivalent to the mechanical *continuity equation*

$$\partial_t \rho + \text{div}(\rho \dot{\mathbf{x}}) = 0. \quad (3.65)$$

Conservative Quantum System

We perform the formal quantization of the conservative equation (3.64) in the Heisenberg picture: all variables become Hermitian operators (denoted by ‘ $\wedge$ ’), the symplectic phase-space $T^*Q = \{q^i, p_i\}$ becomes the Hilbert state-space $\mathcal{H} = \mathcal{H}_{\hat{q}^i} \otimes \mathcal{H}_{\hat{p}_i}$ (where $\mathcal{H}_{\hat{q}^i} = \mathcal{H}_{\hat{q}^1} \otimes \dots \otimes \mathcal{H}_{\hat{q}^N}$ and $\mathcal{H}_{\hat{p}_i} = \mathcal{H}_{\hat{p}_1} \otimes \dots \otimes \mathcal{H}_{\hat{p}_N}$), the classical Poisson bracket $[\cdot, \cdot]$ becomes the quantum commutator $\{\cdot, \cdot\}$ multiplied by $-i/\hbar$

$$[\cdot, \cdot] \longrightarrow -i\{\cdot, \cdot\} \quad (\hbar = 1 \text{ in normal units}). \quad (3.66)$$

In this way the classical Liouville equation (3.64) becomes the *quantum Liouville equation*

$$\partial_t \hat{\rho} = i\{\hat{\rho}, \hat{H}\}, \quad (3.67)$$

where $\hat{H} = \hat{H}(\hat{q}^i, \hat{p}_i, t)$ is the Hamiltonian evolution operator, while

$$\hat{\rho} = P(a)|\Psi_a\rangle\langle\Psi_a|, \quad \text{with} \quad \text{Tr}(\hat{\rho}) = 1,$$

denotes the von Neumann *density matrix operator*, where each quantum state $|\Psi_a\rangle$ occurs with probability $P(a)$; $\hat{\rho} = \hat{\rho}(\hat{q}^i, \hat{p}_i, t)$ is closely related to another von Neumann concept: *entropy* $S = -\text{Tr}(\hat{\rho} \ln \hat{\rho})$.

Open Classical System

We now move to the *open (nonconservative) system*: on the macroscopic NN-level the *opening operation* equals to the *adding* of a *covariant* vector of external (dissipative and/or motor) forces $F_i = F_i(q^i, p_i, t)$ to (the r.h.s of) the covariant Hamiltonian *force equation*, so that Hamiltonian equations get the *open (dissipative and/or forced) form*

$$\dot{q}^i = \frac{\partial H}{\partial p_i}, \quad \dot{p}_i = F_i - \frac{\partial H}{\partial q^i}. \quad (3.68)$$

In the framework of the open Hamiltonian system (3.68), dynamics of any function $A(q^i, p_i, t)$ is defined by the *open evolution equation*:

$$\partial_t A = -[A, H] + \Phi,$$

where $\Phi = \Phi(F_i)$ represents the general form of the scalar force term.

In particular, if $A = \rho(q^i, p_i, t)$ represents the density function of ensemble phase-points, then its dynamics is given by the (dissipative/forced) *open Liouville equation*:

$$\partial_t \rho = -[\rho, H] + \Phi. \quad (3.69)$$

In particular, the scalar force term can be cast as a linear Poisson-bracket form

$$\Phi = F_i[A, q^i], \quad \text{with} \quad [A, q^i] = -\frac{\partial A}{\partial p_i}. \quad (3.70)$$

Now, in a similar way as the conservative Liouville equation (3.64) resembles the continuity equation (3.65) from continuum dynamics, also the open Liouville equation (3.69) resembles the probabilistic *Fokker-Planck equation* from statistical mechanics. If we have a ND stochastic process $x(t) = (q^i(t), p_i(t))$ defined by the vector *Itô SDE*

$$dx(t) = f(x, t) dt + G(x, t) dW,$$

where f is a ND vector function, W is a KD Wiener process, and G is a $N \times KD$ matrix valued function, then the corresponding probability density function $\rho = \rho(x, t | \dot{x}, t')$ is defined by the ND Fokker-Planck equation (see, e.g., [Gar94])

$$\partial_t \rho = -\text{div}[\rho f(x, t)] + \frac{1}{2} \frac{\partial^2}{\partial x_i \partial x_j} (Q_{ij} \rho), \quad (3.71)$$

where $Q_{ij} = (G(x, t) G^T(x, t))_{ij}$. It is obvious that the Fokker-Planck equation (3.71) represents the particular, stochastic form of our general open Liouville equation (3.69), in which the scalar force term is given by the (second-derivative) noise term

$$\Phi = \frac{1}{2} \frac{\partial^2}{\partial x_i \partial x_j} (Q_{ij} \rho).$$

Equation (3.69) will represent the open classical model of our macroscopic NN-dynamics.

Continuous Neural Network Dynamics

The generalized NN-dynamics, including two special cases of *graded response neurons* (GRN) and *coupled neural oscillators* (CNO), can be presented in the form of a stochastic *Langevin rate equation*

$$\dot{\sigma}_i = f_i + \eta_i(t), \quad (3.72)$$

where $\sigma_i = \sigma_i(t)$ are the continual neuronal variables of i th neurons (representing either membrane action potentials in case of GRN, or oscillator phases in case of CNO); J_{ij} are individual synaptic weights; $f_i = f_i(\sigma_i, J_{ij})$ are the deterministic forces (given, in GRN-case, by $f_i = \sum_j J_{ij} \tanh[\gamma \sigma_j] - \sigma_i + \theta_i$, with $\gamma > 0$ and with the θ_i representing injected currents, and in CNO-case, by $f_i = \sum_j J_{ij} \sin(\sigma_j - \sigma_i) + \omega_i$, with ω_i representing the natural frequencies of the individual oscillators); the noise variables are given as $\eta_i(t) = \lim_{\Delta \rightarrow 0} \zeta_i(t) \sqrt{2T/\Delta}$ where $\zeta_i(t)$ denote uncorrelated Gaussian distributed random forces and the parameter T controls the amount of noise in the system, ranging from $T = 0$ (deterministic dynamics) to $T = \infty$ (completely random dynamics).

More convenient description of the neural random process (3.72) is provided by the Fokker–Planck equation describing the time evolution of the probability density $P(\sigma_i)$

$$\partial_t P(\sigma_i) = -\frac{\partial}{\partial \sigma_i} (f_i P(\sigma_i)) + T \frac{\partial^2}{\partial \sigma_i^2} P(\sigma_i). \quad (3.73)$$

Now, in the case of deterministic dynamics $T = 0$, equation (3.73) can be put into the form of the conservative Liouville equation (3.64), by making the substitutions: $P(\sigma_i) \rightarrow \rho$, $f_i = \dot{\sigma}_i$, and $[\rho, H] = \text{div}(\rho \dot{\sigma}_i) \equiv \sum_i \frac{\partial}{\partial \sigma_i} (\rho \dot{\sigma}_i)$, where $H = H(\sigma_i, J_{ij})$. Further, we can formally identify the stochastic forces, i.e., the second-order noise-term $T \sum_i \frac{\partial^2}{\partial \sigma_i^2} \rho$ with $F^i[\rho, \sigma_i]$, to get the open Liouville equation (3.69).

Therefore, on the NN-level deterministic dynamics corresponds to the conservative system (3.64). Inclusion of stochastic forces corresponds to the system opening (3.69), implying the *macroscopic arrow of time*.

Open Quantum System

By formal quantization of equation (3.69) with the scalar force term defined by (3.70), in the same way as in the case of the conservative dynamics, we get the *quantum open Liouville equation*

$$\partial_t \hat{\rho} = i\{\hat{\rho}, \hat{H}\} + \hat{\Phi}, \quad \text{with} \quad \hat{\Phi} = -i\hat{F}_i\{\hat{\rho}, \hat{q}^i\}, \quad (3.74)$$

where $\hat{F}_i = \hat{F}_i(\hat{q}^i, \hat{p}_i, t)$ represents the covariant quantum operator of external friction forces in the Hilbert state-space $\mathcal{H} = \mathcal{H}_{\hat{q}^i} \otimes \mathcal{H}_{\hat{p}_i}$.

Equation (3.74) will represent the open quantum-friction model of our microscopic MT-dynamics. Its system-independent properties are [EMN92, EMN99, MN95a, MN95b, Nan95]:

1. Conservation of probability P

$$\partial_t P = \partial_t [\text{Tr}(\hat{\rho})] = 0.$$

2. Conservation of energy E , on the average

$$\partial_t \langle \langle E \rangle \rangle \equiv \partial_t [\text{Tr}(\hat{\rho} E)] = 0.$$

3. Monotonic increase in entropy

$$\partial_t S = \partial_t [-\text{Tr}(\hat{\rho} \ln \hat{\rho})] \geq 0,$$

and thus automatically and naturally implies a *microscopic arrow of time*, so essential in realistic biophysics of neural processes.

Non-Critical Stringy MT-Dynamics

In EMN-language of non-critical (SUSY) bosonic strings, our MT-dynamics equation (3.74) reads

$$\partial_t \hat{\rho} = i\{\hat{\rho}, \hat{H}\} - i\hat{g}_{ij}\{\hat{\rho}, \hat{q}^i\}\hat{q}^j, \quad (3.75)$$

where the target-space density matrix $\hat{\rho}(\hat{q}^i, \hat{p}_i)$ is viewed as a function of coordinates $\hat{q}^i$ that parameterize the couplings of the generalized σ -models on the bosonic string world-sheet, and their conjugate momenta $\hat{p}_i$, while $\hat{g}_{ij} = \hat{g}_{ij}(\hat{q}^i)$ is the quantum operator of the *positive definite metric* in the space of couplings. Therefore, the covariant quantum operator of external friction forces is in EMN-formulation given as $\hat{F}_i(\hat{q}^i, \hat{q}^i) = \hat{g}_{ij} \hat{q}^j$.

Equation (3.75) establishes the conditions under which a large-scale coherent state appearing in the MT-network, which can be considered responsible for loss-free energy transfer along the tubulins.

Equivalence of Neurodynamic Forms

It is obvious that both the macroscopic NN-equation (3.69) and the microscopic MT-equation (3.74) have the same open Liouville form, which implies the arrow of time. These proves the existence of the formal neuro-biological space-time self-similarity.

In this way, we have described neurodynamics of both NN and MT ensembles, belonging to completely different biophysical space-time scales, by the unique form of open Liouville equation, which implies the arrow of time. The existence of the formal *neuro-biological self-similarity* has been proved.

Dissipative Quantum Brain Model

The *conservative brain* model was originally formulated within the framework of the quantum field theory (QFT) by [RU67] and subsequently developed in [STU78, STU79, JY95, JPY96]. The conservative brain model has been recently extended to the *dissipative quantum dynamics* in the work of G. Vitiello and collaborators [Vit95, AV00, PV99, Vit01, PV03, PV04].

The canonical quantization procedure of a dissipative system requires to include in the formalism also the system representing the environment (usually the heat bath) in which the system is embedded. One possible way to do that is to depict the environment as the time-reversal image of the system [CRV92]: the environment is thus described as the *double* of the system in the time-reversed dynamics (the system image in the mirror of time).

Within the framework of dissipative QFT, the brain system is described in terms of an *infinite collection of damped harmonic oscillators* A_κ (the simplest prototype of a dissipative system) representing the DWQ [Vit95]. Now, the collection of damped harmonic oscillators is ruled by the Hamiltonian [Vit95, CRV92]

$$H = H_0 + H_I, \quad \text{with} \\ H_0 = \hbar\Omega_\kappa(A_\kappa^\dagger A_\kappa - \tilde{A}_\kappa^\dagger \tilde{A}_\kappa), \quad H_I = i\hbar\Gamma_\kappa(A_\kappa^\dagger \tilde{A}_\kappa^\dagger - A_\kappa \tilde{A}_\kappa),$$

where Ω_κ is the frequency and Γ_κ is the damping constant. The $\tilde{A}_\kappa$ modes are the ‘time-reversed mirror image’ (i.e., the ‘mirror modes’) of the A_κ modes. They are the doubled modes, representing the environment modes, in such a way that κ generically labels their degrees of freedom. In particular, we consider the damped harmonic oscillator (DHO)

$$m\ddot{x} + \gamma\dot{x} + \kappa x = 0, \quad (3.76)$$

as a simple prototype for dissipative systems (with intention that thus get results also apply to more general systems). The damped oscillator (3.76) is a non-Hamiltonian system and therefore the customary canonical quantization procedure cannot be followed. However, one can face the problem by resorting to well known tools such as the *density matrix* ρ and the *Wigner function* W .

Let us start with the special case of a *conservative particle* in the absence of friction γ , with the standard Hamiltonian, $H = -(\hbar\partial_x)^2/2m + V(x)$.

Recall (from the previous subsection) that the *density matrix equation of motion*, i.e., *quantum Liouville equation*, is given by

$$i\hbar\dot{\rho} = [H, \rho]. \quad (3.77)$$

The density matrix function ρ is defined by

$$\langle x + \frac{1}{2}y | \rho(t) | x - \frac{1}{2}y \rangle = \psi^*(x + \frac{1}{2}y, t) \psi(x - \frac{1}{2}y, t) \equiv W(x, y, t),$$

with the associated standard expression for the *Wigner function* (see [FH65]),

$$W(p, x, t) = \frac{1}{2\pi\hbar} \int W(x, y, t) e^{(-i\frac{py}{\hbar})} dy.$$

Now, in the coordinate x -representation, by introducing the notation

$$x_\pm = x \pm \frac{1}{2}y, \quad (3.78)$$

the Liouville equation (3.77) can be expanded as

$$\begin{aligned} i\hbar \partial_t \langle x_+ | \rho(t) | x_- \rangle = \\ \left\{ -\frac{\hbar^2}{2m} [\partial_{x_+}^2 - \partial_{x_-}^2] + [V(x_+) - V(x_-)] \right\} \langle x_+ | \rho(t) | x_- \rangle, \end{aligned} \quad (3.79)$$

while the Wigner function $W(p, x, t)$ is now given by

$$\begin{aligned} i\hbar \partial_t W(x, y, t) &= H_o W(x, y, t), \quad \text{with} \\ H_o &= \frac{1}{m} p_x p_y + V(x + \frac{1}{2}y) - V(x - \frac{1}{2}y), \\ \text{and} \quad p_x &= -i\hbar \partial_x, \quad p_y = -i\hbar \partial_y. \end{aligned} \quad (3.80)$$

The new Hamiltonian H_o (3.80) may be get from the corresponding Lagrangian

$$L_o = m\dot{x}\dot{y} - V(x + \frac{1}{2}y) + V(x - \frac{1}{2}y). \quad (3.81)$$

In this way, Vitiello concluded that the density matrix and the Wigner function formalism *required*, even in the conservative case (with zero mechanical resistance γ), the introduction of a ‘doubled’ set of coordinates, $x_{\pm}$, or, alternatively, x and y . One may understand this as related to the introduction of the ‘couple’ of indices *necessary* to label the density matrix elements (3.79).

Let us now consider the case of the *particle interacting* with a *thermal bath* at temperature T . Let f denote the *random force* on the particle at the position x due to the bath. The interaction Hamiltonian between the bath and the particle is written as

$$H_{int} = -fx. \quad (3.82)$$

Now, in the *Feynman–Vernon formalism* (see [Fey72]), the *effective action* $A[x, y]$ for the particle is given by

$$A[x, y] = \int_{t_i}^{t_f} L_o(\dot{x}, \dot{y}, x, y) dt + I[x, y],$$

with L_o defined by (3.81) and

$$e^{\frac{i}{\hbar} I[x, y]} = \langle (e^{-\frac{i}{\hbar} \int_{t_i}^{t_f} f(t)x_-(t)dt})_- (e^{\frac{i}{\hbar} \int_{t_i}^{t_f} f(t)x_+(t)dt})_+ \rangle, \quad (3.83)$$

where the symbol $\langle \cdot \rangle$ denotes the average with respect to the thermal bath; ‘ $(\cdot)_+$ ’ and ‘ $(\cdot)_-$ ’ denote time ordering and anti-time ordering, respectively; the coordinates $x_{\pm}$ are defined as in (3.78). If the interaction between the bath and the coordinate x (3.82) were turned off, then the operator f of the bath would develop in time according to $f(t) = e^{iH_{\gamma}t/\hbar} f e^{-iH_{\gamma}t/\hbar}$, where H_{γ} is the Hamiltonian of the isolated bath (decoupled from the coordinate x). $f(t)$ is then the force operator of the bath to be used in (3.83).

The interaction $I[x, y]$ between the bath and the particle has been evaluated in [SVW95] for a linear passive damping due to thermal bath by following Feynman–Vernon and Schwinger [FH65]. The final result from [SVW95] is:

$$I[x, y] = \frac{1}{2} \int_{t_i}^{t_f} dt [x(t) F_y^{ret}(t) + y(t) F_x^{adv}(t)] \\ + \frac{i}{2\hbar} \int_{t_i}^{t_f} \int_{t_i}^{t_f} dt ds N(t-s) y(t) y(s),$$

where the retarded force on y , F_y^{ret} , and the advanced force on x , F_x^{adv} , are given in terms of the retarded and advanced Green functions $G_{ret}(t-s)$ and $G_{adv}(t-s)$ by

$$F_y^{ret}(t) = \int_{t_i}^{t_f} ds G_{ret}(t-s) y(s), \quad F_x^{adv}(t) = \int_{t_i}^{t_f} ds G_{adv}(t-s) x(s),$$

respectively. In (3.84), $N(t-s)$ is the *quantum noise* in the fluctuating random force given by: $N(t-s) = \frac{1}{2} \langle f(t)f(s) + f(s)f(t) \rangle$.

The real and the imaginary part of the action are given respectively by

$$\text{Re}(A[x, y]) = \int_{t_i}^{t_f} L dt, \quad (3.84)$$

$$L = m\dot{x}\dot{y} - \left[V(x + \frac{1}{2}y) - V(x - \frac{1}{2}y) \right] + \frac{1}{2} [x F_y^{ret} + y F_x^{adv}], \quad (3.85)$$

$$\text{and} \quad \text{Im}(A[x, y]) = \frac{1}{2\hbar} \int_{t_i}^{t_f} \int_{t_i}^{t_f} N(t-s) y(t) y(s) dt ds. \quad (3.86)$$

Equations (3.84–3.86), are *exact* results for linear passive damping due to the bath. They show that in the classical limit ‘ $\hbar \rightarrow 0$ ’ nonzero y yields an ‘unlikely process’ in view of the large imaginary part of the action implicit in (3.86). Nonzero y , indeed, may lead to a negative real exponent in the evolution operator, which in the limit $\hbar \rightarrow 0$ may produce a negligible contribution to the probability amplitude. On the contrary, at quantum level nonzero y accounts for quantum noise effects in the fluctuating random force in the system–environment coupling arising from the imaginary part of the action (see [SVW95]).

When in (3.85) we use

$$F_y^{ret} = \gamma \dot{y} \quad \text{and} \quad F_x^{adv} = -\gamma \dot{x} \quad \text{we get,}$$

$$L(\dot{x}, \dot{y}, x, y) = m\dot{x}\dot{y} - V\left(x + \frac{1}{2}y\right) + V\left(x - \frac{1}{2}y\right) + \frac{\gamma}{2}(x\dot{y} - y\dot{x}). \quad (3.87)$$

By using

$$V\left(x \pm \frac{1}{2}y\right) = \frac{1}{2}\kappa\left(x \pm \frac{1}{2}y\right)^2$$

in (3.87), the DHO equation (3.76) and its complementary equation for the y coordinate

$$m\ddot{y} - \gamma\dot{y} + \kappa y = 0. \quad (3.88)$$

are derived. The y -oscillator is the time-reversed image of the x -oscillator (3.76). From the manifolds of solutions to equations (3.76) and (3.88), we could choose those for which the y coordinate is constrained to be zero, they simplify to

$$m\ddot{x} + \gamma\dot{x} + \kappa x = 0, \quad y = 0.$$

Thus we get the classical damped oscillator equation from a Lagrangian theory at the expense of introducing an ‘extra’ coordinate y , later constrained to vanish. Note that the constraint $y(t) = 0$ is *not* in violation of the equations of motion since it is a true solution to (3.76) and (3.88).

3.2.2 Macroscopic Neurodynamics

Hopfield’s Neural Nets

Ising-Spin Hopfield Neurons

Recall that Hopfield [Hop82] gives a collection of simple threshold automata, called formal neurons by McCulloch and Pitts: two-state, ‘all-or-none’, firing or non-firing units that can be modeled by Ising spins (uniaxial magnets) $\{S_i\}$ such that $S_i = \pm 1$ (where $1 = |\uparrow\rangle$ –‘spin up’ and $-1 = |\downarrow\rangle$ –‘spin down’; the label of the neuron is i and ranges between 1 and the size of the network N). The neurons are connected by synapses J_{ij} .

A (firing) patterns $\{\xi_i^\mu\}$ represent specific S_i -spin configurations (where the label of the pattern is μ and ranges between 1 and q).

Using random patterns $\xi_i^\mu = \pm 1$ with equal probability $1/2$, we have the synaptic efficacy J_{ij} of j th neuron operating on i th neuron given by

$$J_{ij} = N^{-1}\xi_i^\mu \xi_j^\mu \equiv N^{-1}\boldsymbol{\xi}_i \cdot \boldsymbol{\xi}_j. \quad (3.89)$$

Postsynaptic potential (PSP) represents an internal local field

$$h_i(t) = J_{ij}S_j(t). \quad (3.90)$$

Now, the sequential (threshold) dynamics is defined in the form of discrete equation

$$S_i(t + \Delta t) = \text{sgn}[h_i(t)]. \quad (3.91)$$

Dynamics (3.91) is equivalent to the rule that the state of a neuron is changed, or a spin is flipped if and only if the total network energy, given by Ising Hamiltonian

$$H_N = -\frac{1}{2} J_{ij} S_i S_j \quad (3.92)$$

is lowered [Hop82, HT85]. Therefore, the Ising Hamiltonian H_N represents the monotonically decreasing Lyapunov function for the sequential dynamics (3.91), which converges to a local minimum or ground state of H_N . This holds for any symmetric coupling $J_{ij} = J_{ji}$ with $J_{ii} = 0$ and if spin-updating in (3.91) is asynchronous. In this case the patterns $\{\xi_i^\mu\}$ after convergence become identical, or very near to, ground states of H_N , each of them at the bottom of the valley.

Data are stored in the neural net if, by a suitable choice of the J_{ij} , several specific patterns $\{\xi_i^\mu\}$ are made local minima of H_N . If this can be achieved, the neural net will function as content-addressable or (auto)associative memory. A network state which ‘somehow resembles’ one of the stored prototypes corresponds to a location in the energy landscape which is close enough to the minimum representing that prototype to lie in its basin of attraction. By spontaneously moving downhill, or relaxing to the energy minimum, the network recalls the data or reconstructs the prototype.

Suppose that we have somehow stored several (stationary) patterns $\{\xi_i^\mu\}$ in the J_{ij} and that the system is offered a noisy version of one of them. If the noise was not too strong, the system remains in the valley associated with that pattern and under its natural dynamics it will relax to the energy minimum where the stored patterns live. That is, the system has recalled the pattern.

In statistical mechanics, one is usually given the synapses J_{ij} and one of the first tasks consists in finding the minima of the Ising Hamiltonian H_N . However, in the theory of neural networks, one is given the patterns $\{\xi_i^\mu\}$ and one is asked to solve the inverse problem: finding synapses J_{ij} such that the patterns $\{\xi_i^\mu\}$ are minima of the Hamiltonian H_N .

To see why the Hopfield model with synapses given by (3.89) has patterns $\{\xi_i^\mu\}$ as attractors of the dynamics (3.91), note that the sequential dynamical law embodies a two-step process, the evolution of the local field (PSP) (3.90), which is a linear operation, and a nonlinear decision process (3.91).

Graded-Response Hopfield Networks

Recall that graded-response neurons have continuous input-output relation (like nonlinear amplifiers) of the form $V_i = g_i(\lambda u_i)$, where u_i denotes the input at i , a constant λ is called the gain parameter, and V_i is the output [Hop84]. Usually, g_i are taken to be sigmoid functions, odd, and monotonically increasing (e.g., $g(\cdot) = \frac{1}{2}(1 + \tanh(\cdot))$), while discrete Ising spins have $g_i(u_i) = \text{sgn}_i(u_i)$. The behavior of the continuous Hopfield neural network is usually described by a set of coupled RC-transient equations

$$C_i \dot{u}_i = I_i + J_{ij} V_j - \frac{u_i}{R_i} \quad (3.93)$$

where $u_i = g^{-1}(V_i)$, R_i and C_i denote input capacitance and resistance, and I_i represents an external source.

The Hamiltonian of the continuous system (3.93) is given by

$$H = -\frac{1}{2}J_{ij}V_iV_j + \sum_{i=1}^N R_i^{-1} \int_0^{V_i} dV g^{-1}(V) - I_i V_i. \quad (3.94)$$

However, according to [Hop84], the synapses J_{ij} retain the form (3.89) with random patterns $\xi_i^\mu = \pm 1$ with equal probability $1/2$, and the synaptic symmetry $J_{ij} = J_{ji}$ implies that the continuous Hamiltonian (3.94) represents a Lyapunov function of the system (3.93), i.e., H decreases under the continual neurodynamics governed by equation (3.93) as time proceeds.

More general form of synapses is

$$J_{ij} = N^{-1}Q(\xi_i; \xi_j),$$

for some synaptic kernel Q on $R^q \times R^q$. The vector ξ_i varies as i travels from 1 to N , but remains on a corner of the Hamming hypercube $[-1, 1]^q$.

Hopfield overlaps

Assuming that the number q of stored patterns is small compared to the number of neurons, i.e., $q/N \rightarrow 0$, we find that the synapses (3.89) give rise to a local field of the form

$$h_i = \xi_i^\mu m_\mu, \quad \text{where} \quad m_\mu = N^{-1} \xi_i^\mu S_i \quad (3.95)$$

is the *auto-overlap* (or simply *overlap*)⁴ of the network state $\{S_i\}$ with the pattern $\{\xi_i^\mu\}$, measuring the proximity between them. We can see that $m_\mu = 1$ (like peak-up in auto-correlation) if $\{S_i\}$ and $\{\xi_i^\mu\}$ are identical patterns, $m_\mu = -1$ (like peak-down in autocorrelation) if they are each other's complement, and $m_\mu = O(1/\sqrt{N})$ if they are uncorrelated (like no-peak in auto-correlation) with each other. Overlaps m_μ are related to the Hamming distance d_μ between the patterns (the fraction of spins which differ) by $d_\mu = \frac{1}{2}(1 - m_\mu)$.

As a pattern ξ_i^μ represents (in the simplest case) a specific Ising-spin S_i -configuration, then $(\xi_i^\mu)^2 = 1$. If $S_i = \xi_i^\mu$ for all i , then $m_\mu = 1$. Conversely, if $m_\mu = 1$, then $S_i = \xi_i^\mu$. In all other cases $m_\mu < 1$, by the Cauchy-Schwartz inequality. If ξ_i^μ and S_i are uncorrelated, we may expect m_μ to be of the order of $N^{-1/2}$, since the sum consists of N terms, each containing a ξ_i^μ . On the other hand, if the S_i are positively correlated with ξ_i^μ , then m_μ is of the order of unity. So the overlaps give the global information about the network and hence are good order parameters, in terms of Haken's synergetics (see [Hak83]). Also, according to [Hop84], the extension to the continual network is straightforward.

⁴ The auto-overlap resembles the auto-correlation function of a time-series, where distinct peaks indicate that the series at the certain time t is similar to the series at time $t + \Delta t$.

Using overlaps, the Ising Hamiltonian becomes

$$H_N = -\frac{1}{2}N \sum_{\mu=1}^q m_{\mu}^2.$$

The similarity between two different patterns ξ_i^{μ} and ξ_i^{ν} is measured by their mutual overlap or *cross-overlap* $m_{\mu\nu}$ (in other parlance it is called Karhunen–Loeve matrix (see [FS92]), which extracts the principal components from a data set)⁵, equal

$$m_{\mu\nu} = N^{-1} \xi_i^{\mu} \xi_i^{\nu}. \quad (3.96)$$

For similar patterns the cross-overlap is close to unity whereas for uncorrelated patterns it is random variable with zero mean and small $(1/\sqrt{N})$ variance.

The symmetric Hopfield synaptic matrix J_{ij} can be expressed in terms of the cross-overlaps $m_{\mu\nu}$ as

$$J_{ij} = N^{-1} \xi_i^{\mu} (m_{\mu\nu})^{-1} \xi_j^{\nu} = J_{ji},$$

where $(m_{\mu\nu})^{-1}$ denotes the *Moore–Penrose pseudoinverse* of the cross-overlap matrix $m_{\mu\nu}$.

Besides the Hopfield model, the proposed pattern-overlap picture can be extended to cover some more sophisticated kinds of associative memory, to two of them:

(i) The so-called *forgetful memories*, characterized by iterative synaptic prescription

$$J_{ij}^{(\mu)} = \phi(\epsilon \xi_i^{\mu} \xi_j^{\mu} + J_{ij}^{(\mu-1)}),$$

for some small parameter ϵ and some odd function ϕ . If $\phi(\cdot)$ saturates as $|\cdot| \rightarrow \infty$, the memory creates storage capacity for new patterns by forgetting the old ones.

(ii) The so-called *temporal associative memories*, which can store and retrieve a sequence of patterns, through synapses

$$NJ_{ij} = \xi_i^{\mu} \xi_j^{\mu} + \epsilon \xi_i^{(\mu+1)} \xi_j^{\mu},$$

where the second term on the right is associated with a temporal delay, so that one can imagine that the second term ‘pushes’ the neural system through an energy landscape created by the first term.

⁵ The cross-overlap resembling the cross-correlation function of two time-series, with several distinct peaks, indicating that the two series are very similar at each point in time where the peaks occur

Noise Delayed Bifurcation in Coupled Neurons

The Theta-Neuron

The *theta-neuron model* developed by Ermentrout and Gutkin [Erm96, Gut98] is derived from the observation that wide class of neuronal models of cortical neurons, based on the electro-physiological model of Hodgkin and Huxley, show a saddle-node type bifurcation at a critical parameter value. This parameter determines the dynamical behavior of the solutions of the corresponding system of ODEs. General dynamical systems theory tells us that the qualitative behavior in some neighborhood of the bifurcation point (which may be quite large as it extends up to the next bifurcation or other dynamic transition) is governed by the reduction of the system to the *center manifold* (see section 2.3 above). In the present case of *the saddle-node bifurcation* which is the simplest bifurcation type, this leads to the ODE: $\dot{x} = \lambda + x^2$. Here, the bifurcation parameter λ is considered as the input to the neuron while x records its activity. Obviously, a solution to this equation tends to infinity in finite time. This is considered as a spiking event, and the initial values are then reset to $-\infty$. In order to have a model that does not exhibit such formal singularities, one introduces a phase variable θ that is 2π -periodic via $x = \tan(\frac{\theta}{2})$. θ is then a variable with domain the unit circle S^1 , and a spike now corresponds to a period of θ . Spikes are no longer represented by transitions through infinity, but by changes of some discrete topological invariant. The original differential equation is then transformed into

$$\dot{\theta} = (1 - \cos \theta) + (1 + \cos \theta)\lambda. \quad (3.97)$$

Due to the nonlinearity of the transformation from x to θ , the input λ is no longer additive. In fact, it is easy to show that $(1 + \cos \theta)$ is the phase resetting function for the model [Erm96]. As before, the bifurcation occurs at $\lambda = 0$. There, we have precisely one rest point, namely $\theta = 0$ which is degenerate. In any case, the sensitivity to the input λ is highest at $\theta = 0$ and lowest at $\theta = \pi$ which according to the derivation of our equation is considered as the spike point. When λ is positive, the equation does not have any rest point. In this case, θ continues to increase all the time, and the neuron is perpetually firing. When λ is negative, however, there are two rest points, a stable one denoted by θ_r and an unstable one $\theta_t > \theta_r$. If θ is larger than θ_t it increases until it completes a period and comes to rest at $\theta_r + 2\pi$ which is identified with θ_r as we are working on the unit circle S^1 . Thus, if the phase is above the threshold value θ_t , a spike occurs and the neuron returns to rest. So far, we have tacitly assumed that the input λ is constant. We now consider the situation where the input can be decomposed as $\lambda = \beta + \sigma\eta$, where β is a constant term, the so-called ‘bias’, while η is (white) noise and σ its intensity. In this case, sufficiently strong noise can occasionally push the phase θ beyond the threshold value θ_t causing intermittent firing. Equation

(3.97) now becomes a canonical stochastic saddle-node oscillator which has been studied in [Gut98].

Coupled Theta-Neurons

Here we consider the situation where we have two neurons (distinguished by subscripts $i = 1, 2$). The dynamics then takes place on the product of two circles, i.e., on a 2D torus T , represented by the square $[-\pi, \pi] \times [-\pi, \pi]$ in the plane, with periodic boundary identifications. We first consider the simple case of two uncoupled, noise-free neurons ($\sigma_1 = \sigma_2 = 0$) with the same bias β . Their dynamics are independent. If $\beta > 0$, both neurons continue to fire, although their phase difference, if not 0 initially, is not constant, due to the nonlinearity of the differential equation governing it. If $\beta = 0$, $(0, 0)$ is a degenerate rest point. The two trajectories $\theta_1 = \theta_2 = 0$ are homoclinic orbits and all flow lines eventually terminate at this fixed-point. One or both neurons will spike before returning to rest if their initial phase is between 0 and π [Gut98].

If $\beta < 0$, we have four fixed-points – the attractor ($\theta_1 = \theta_2 = \theta_r$), the repeller ($\theta_1 = \theta_2 = \theta_t$), and the two saddles where one of the neurons has its phase at θ_r (rest) and the other one at θ_t (threshold). Some special heteroclinic orbits are given by the straight lines where one of the two neurons stays at θ_t while the other one moves from the threshold to the rest value, spiking if its initial phase was above threshold. All other flow lines terminate at the attractor. We now add an interaction term $s_i g_s$ to the input of neuron i . s_i is considered as the synaptic input from neuron j to neuron i ($i \neq j$) and g_s is the synaptic intensity. (One could also study the case of a single neuron i for which s_i represents synaptic self-coupling, but here we are interested in the case of two coupled neurons). A precise equation for s_i can be derived from electrophysiological models, however for our qualitative study we only need the characteristic features that it stays bounded between 0 and 1. Typically, it is peaked near the spike of neuron j , i.e., where $\theta_j = \pi$. With this interaction term, the equation for neuron i then becomes

$$\dot{\theta}_i = (1 - \cos(\theta_i)) + (1 + \cos(\theta_i))(\beta + g_s s_i + \sigma \eta).$$

Since s_i represents the input that neuron i receives from neuron j , s_i should essentially be considered as a function of the phase θ_j of j . Once more, we first consider the situation without noise, i.e., $\sigma = 0$ (although our final aim is to understand the effect of noise on the dynamic behavior of the coupled neurons). We also assume that we are in the excitable region, i.e., $\beta < 0$. g_s is assumed to be positive (excitatory coupling), and so the coupling counteracts the effect of the bias to a certain extent, a crucial difference being, however, that the synaptic input to each neuron is time-dependent, in contrast to the constant bias. If g_s is sufficiently small, the qualitative situation does not change compared to the case without coupling, i.e., $g_s = 0$. We still have a heteroclinic orbit from the saddle ($\theta_1 = \theta_t, \theta_2 = \theta_r$) to the attractor (θ_r, θ_r),

although θ_2 does not stay constant anymore along that orbit, but increases first a little due to the input from neuron 1 before it descends again to the rest value [Gut98].

If g_s reaches some critical value g_s^* , however, the heteroclinic orbit starting at (θ_t, θ_r) does not terminate anymore at the attractor, and the value of the phase of neuron 2 is increased so much by the synaptic interaction that it reaches the other saddle (θ_r, θ_t) . Besides two heteroclinic orbits that go from the repeller to the two saddles as before, all other orbits still terminate at the attractor (θ_r, θ_r) , for $g_s = g_s^*$. If g_s is increased beyond g_s^* , however, the heteroclinic orbit between the two saddles mutates into a stable attractor. It corresponds to sustained asynchronous firing of the two neurons. In fact, if the phase difference between the two neurons is too small, the dynamics converges towards the double rest point (except in some region in the vicinity of the node), and both neurons stop firing. This is caused by the fact that when the two neurons are close to synchrony, neither cell is sensitive enough to its synaptic input to maintain firing (an effective refractory period). Conversely, if they are out of synchrony, a single spike can induce the second neuron to fire at a time when the first one is close to rest, and sensitive to synaptic input itself. If g_s is only slightly above the critical value, the basin of attraction of that limit cycle will still be relatively small, but as g_s is increased further, the basin grows in size until eventually it is larger than the basin of attraction of the double rest point. On the basis of the preceding analysis, it is now straightforward to predict the effect of noise. If g_s is only slightly above the critical value g_s^* , a small amount of noise is more likely to kick the dynamics out of the narrow basin of attraction of the asynchronous limit cycle and into the large basin of the double rest point than vice versa. In effect, a small noise level increases the critical parameter value required for the qualitative transition to sustained asynchronous firing. A larger amount of noise, however, has the potential to move the dynamics from the rest point into the basin of attraction of the asynchronous limit cycle. Once in that basin, the neurons will fire. Thus, for large noise in that regime, one will observe that the neurons will fire, perhaps with some intermissions spent near the double rest point. So, a larger value of noise will cause intermittent periods of sustained firing of the two neurons even at somewhat smaller values of g_s . In effect, it decreases the value of the critical parameter. Thus, we observe a genuinely nonlinear effect of the noise level σ . For values of the coupling g_s that are substantially larger than the critical value g_s^* , even small amounts of noise have a good chance of perturbing the dynamics out of the attracting vicinity of the double rest point into the attracting region of the asynchronous limit cycle. This will further enhance the sustained asynchronous firing pattern of the two neurons [Gut98].

Spiking Neural Nets

Recently, Izhikevich [Izh04] discussed biological plausibility and computational efficiency of some of the most useful models of *spiking and bursting neurons* (see Figure 3.1). He compared their applicability to large-scale simulations of cortical neural networks.

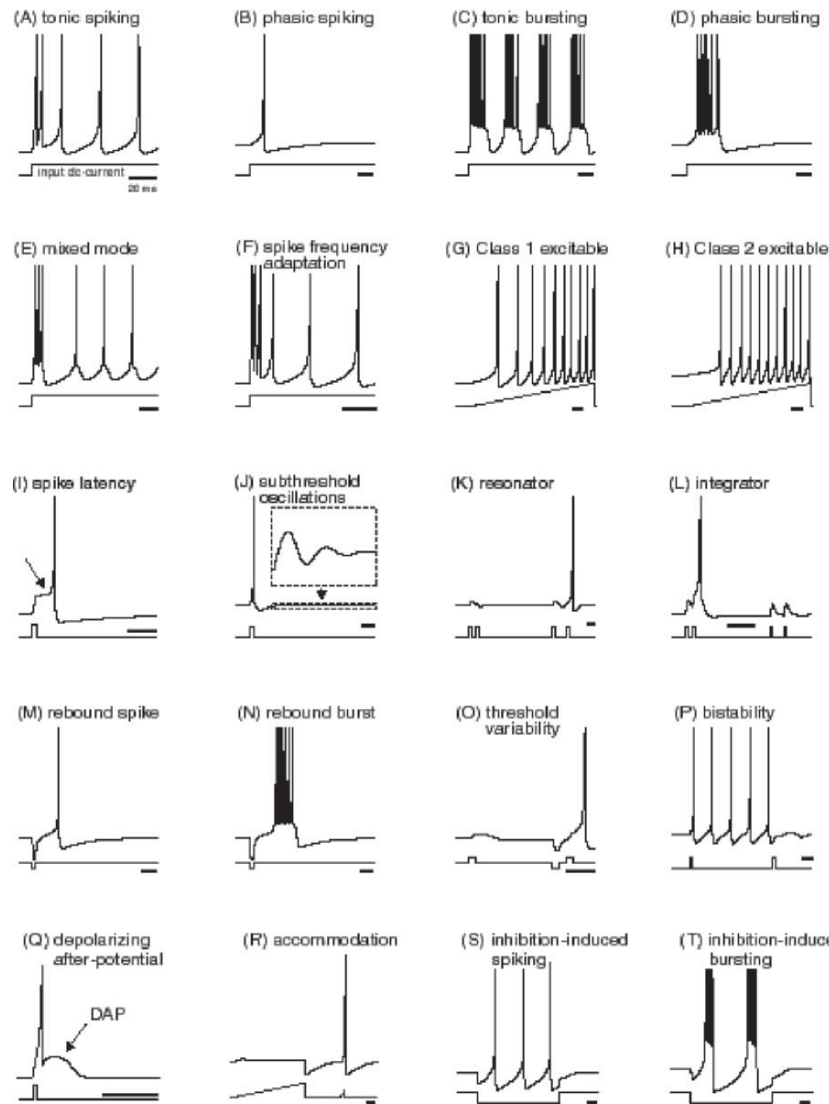


Fig. 3.1. Neuro-computational features of biological neurons (with permission from E. Izhikevich).

Following [Izh04], we present some widely used models of spiking and bursting neurons that can be expressed in the form of ODEs. Throughout this subsection, v denotes the membrane potential. All the parameters in the models are chosen so that v has mV scale and the time has ms scale. To compare computational cost, we assume that each model, written as a dynamical system $\dot{x} = f(x)$, is implemented using the simplest, fixed-step first-order Euler method, with the integration time step chosen to achieve a reasonable numerical accuracy.

Integrate-and-Fire Neuron

One of the most widely used models in computational neuroscience is the *leaky integrate-and-fire neuron*, (I&F neuron, for short) given by

$$\dot{v} = I + a - bv, \quad \text{If } v \geq v_{trsh} \quad \text{Then } v \leftarrow c,$$

where v is the membrane potential, I is the input current, and a, b, c , and v_{trsh} are the parameters. When the membrane potential v reaches the threshold value v_{trsh} , the neuron is said to fire a *spike*, and v is reset to c . The I&F neuron can fire tonic spikes with constant frequency, and it is an integrator. The I&F neuron is *Class 1 excitable system* [Izh99a]; it can fire tonic spikes with constant frequency, and it is an integrator. It is the simplest model to implement when the integration time step τ is 1 ms . Because I&F has only one variable, it cannot have phasic spiking, bursting of any kind, rebound responses, threshold variability, bistability of attractors, or autonomous chaotic dynamics. Because of the fixed threshold, the spikes do not have latencies. In summary, despite its simplicity, I&F is one of the worst models to use in simulations, unless one wants to prove analytical results [Izh04].

Integrate-and-Fire Neuron with Adaptation

The I&F model is 1D, hence it cannot burst or have other properties of cortical neurons. One may think that having a second linear equation

$$\dot{v} = I + a - bv + g(d - v), \quad \dot{g} = (e\delta(t) - g)/\tau,$$

describing activation dynamics of a high-threshold K -current, can make an improvement, e.g., endow the model with spike-frequency adaptation. Indeed, each firing increases the K -activation gate via Dirac δ -function and produces an outward current that slows down the frequency of tonic spiking. This model is fast, yet still lacks many important properties of cortical spiking neurons.

Integrate-and-Fire-or-Burst Neuron

The *integrate-and-fire-or-burst neuron* model is given by

$$\begin{aligned} \dot{v} &= I + a - bv + gH(v - v_h)h(v_T - v), \\ \text{If } v \geq v_{trsh} \quad \text{Then } v &\leftarrow c, \quad \dot{h} = \begin{cases} \frac{-h}{\tau^-}, & \text{if } v > v_h, \\ \frac{1-h}{\tau^+}, & \text{if } v < v_h \end{cases} \end{aligned}$$

to model thalamo-cortical neurons. Here h describes the inactivation of the calcium T -current, g, v_h, v_T, τ^+ and τ^- are parameters describing dynamics of the T -current, and H is the Heaviside step function. Having this kind of a second variable creates the possibility for bursting and other interesting regimes [Izh04], but is already a much slower (depending on the value of v).

Complex-Valued Resonate-and-Fire Neuron

The *resonate-and-fire neuron* is a complex-valued (i.e., $2D$) analogue of the I&F neuron [Izh01], given by

$$\dot{z} = I + (b + iw)z, \quad \text{if } \text{Im } z = a_{trsh} \text{ then } z \leftarrow z_0(z), \quad (3.98)$$

where $z = x + iy \in \mathbb{C}$ is a complex-valued variable that describes oscillatory activity of the neuron. Here b, w , and a_{trsh} are parameters, $i = \sqrt{-1}$, and $z_0(z)$ is an arbitrary function describing activity-dependent after-spike reset. (3.98) is equivalent to the linear system

$$\dot{x} = bx - wy, \quad \dot{y} = wx + by,$$

where the real part x is the current-like variable, while the imaginary part y is the voltage-like variable. The resonate-and-fire model is simple and efficient. When the frequency of oscillation $w = 0$, it becomes an integrator.

Quadratic Integrate-and-Fire Neuron

An alternative to the leaky I&F neuron is the *quadratic I&F neuron*, also known as the *theta-neuron*, or the Ermentrout-Kopell canonical model [Erm96, Gut98]. It can be presented as

$$\dot{v} = I + a(v - v_{rest})(v - v_{trsh}), \quad \text{If } v = v_{trsh} \text{ Then } v \leftarrow v_{rest},$$

where v_{rest} and v_{trsh} are the resting and threshold values of the membrane potential. This model is canonical in the sense that any *Class 1 excitable system* [Izh99a] described by smooth ODEs can be transformed into this form by a continuous change of variables. It takes only seven operations to simulate 1 ms of the model, and this should be the model of choice when one simulates large-scale networks of integrators. Unlike its linear analogue, the quadratic I&F neuron has spike latencies, activity dependent threshold (which is v_{trsh} only when $I = 0$), and bistability of resting and tonic spiking modes.

FitzHugh-Nagumo Neuron

The parameters in the *FitzHugh-Nagumo neuron* model

$$\dot{v} = a + bv + cv^2 + dv^3 - u, \quad \dot{u} = \varepsilon(ev - u),$$

can be tuned so that the model describes spiking dynamics of many resonator neurons. Since one needs to simulate the shape of each spike, the time step

in the model must be relatively small, e.g., $\tau = 0.25 \text{ ms}$. Since the model is a $2D$ system of ODEs, without a reset, it cannot exhibit autonomous chaotic dynamics or bursting. Adding noise to this, or some other $2D$ models, allows for stochastic bursting.

Hindmarsh–Rose Neuron

The *Hindmarsh–Rose thalamic neuron* model [RH89] can be written as a $3D$ ODE system

$$\dot{v} = I + u - F(v) - w, \quad \dot{u} = G(v) - u, \quad \dot{w} = (H(v) - w)/\tau,$$

where F, G , and H are some functions. This model is quite expensive to implement as a large-scale spike simulator [Izh04].

Morris–Lecar Neuron

Morris and Lecar [ML81] suggested a simple $2D$ model to describe oscillations in barnacle giant muscle fiber. Because it has biophysically meaningful and measurable parameters, the *Morris–Lecar neuron* model became quite popular in computational neuroscience community. It consists of a membrane potential equation with instantaneous activation of Ca current and an additional equation describing slower activation of K current,

$$\begin{aligned} C\dot{V} &= I - g_L(V - V_L) - g_{Ca}m_\infty(V)(V - V_{Ca}) - g_Kn(V - V_K), \\ \dot{n} &= \lambda(V)(n_\infty(V) - n), \quad \text{where} \\ m_\infty(V) &= \frac{1}{2} \left(1 + \tanh \left[\frac{V - V_1}{V_2} \right] \right), \quad \text{and} \\ n_\infty(V) &= \frac{1}{2} \left(1 + \tanh \left[\frac{V - V_3}{V_4} \right] \right), \quad \lambda(V) = \bar{\lambda} \cosh \left[\frac{V - V_3}{2V_4} \right], \end{aligned}$$

with parameters: $C = 20 \mu\text{F}/\text{cm}^2$, $g_L = 2 \text{ mmho}/\text{cm}^2$, $V_L = -50 \text{ mV}$, $g_{Ca} = 4 \text{ mmho}/\text{cm}^2$, $V_{Ca} = 10 \text{ mV}$, $g_K = 8 \text{ mmho}/\text{cm}^2$, $V_K = -70 \text{ mV}$, $V_1 = 0 \text{ mV}$, $V_2 = 15 \text{ mV}$, $V_3 = 10 \text{ mV}$, $V_4 = 10 \text{ mV}$, $\bar{\lambda} = 0.1 \text{ s}^{-1}$, and applied current $I(\mu\text{A}/\text{cm}^2)$. The model can exhibit various types of spiking, but could exhibit tonic bursting only when an additional equation is added, e.g., slow inactivation of Ca current. In this case, the model becomes equivalent to the *Hodgkin–Huxley neuron model* [HH52, Hod64], which is extremely expensive to implement.

Activation Dynamics of Graded-Response Networks

To design neurodynamical classifier activation dynamics, we start from the fully recurrent, ND, RC transient circuit, i.e., a nonlinear vector differential equation [Hay94, Kos92, Iva95]:

$$C_j \dot{v}_j = I_j - \frac{v_j}{R_j} + w_{ij} f_i(v_i), \quad (i, j = 1, \dots, N), \quad (3.99)$$

where $v_j = v_j(t)$ represent the activation potentials in the j th neuron, C_j and R_j denote input capacitances and leakage resistances, synaptic weights w_{ij} represent conductances, I_j represent the total currents flowing toward the input nodes, and the functions f_i are sigmoidal.

Geometrically, equation (3.99) defines a smooth autonomous vector-field $X(t)$ in ND neurodynamical phase-space manifold M , and its (numerical) solution for the given initial potentials $v_j(0)$ defines the autonomous neurodynamical phase-flow $\Phi(t) : v_j(0) \rightarrow v_j(t)$ on M .

In AI parlance, equation (3.99) represents a generalization of three well-known recurrent NN models [Hay94, Kos92]:

- (i) continuous Hopfield model,
- (ii) Grossberg ART-family cognitive system, and
- (iii) Hecht-Nielsen counter-propagation network.

Physiologically, equation (3.99) is based on the Nobel-awarded *Hodgkin-Huxley equation* of the neural action potential (for the single squid giant axon membrane) as a function of the conductances g of sodium, potassium and leakage [HH52, Hod64]:

$$C \dot{v} = I(t) - g_{Na}(v - v_{Na}) - g_K(v - v_K) - g_L(v - v_L),$$

where bracket terms represent the electromotive forces acting on the ions.

The continuous Hopfield circuit model:

$$C_j \dot{v}_j = I_j - \frac{v_j}{R_j} + T_{ij} u_i, \quad (i, j = 1, \dots, N), \quad (3.100)$$

where u_i are output functions from processing elements, and T_{ij} is the inverse of the resistors connection-matrix becomes equation (3.99) if we put $T_{ij} = w_{ij}$ and $u_i = f_i[v_j(t)]$.

The Grossberg analogous ART2 system is governed by activation equation:

$$\varepsilon \dot{v}_j = -A v_j + (1 - B v_j) I_j^+ - (C + D v_j) I_j^-, \quad (j = 1, \dots, N),$$

where A, B, C, D are positive constants (A is dimensionally conductance), $0 \leq \varepsilon \ll 1$ is the fast-variable factor (dimensionally capacitance), and I_j^+, I_j^- are excitatory and inhibitory inputs to the j th processing unit, respectively.

General Cohen-Grossberg activation equations [CG83] have the form:

$$\dot{v}_j = -a_j(v_j)[b_j(v_j) - f_k(v_k)m_{jk}], \quad (j = 1, \dots, N), \quad (3.101)$$

and the Cohen-Grossberg theorem ensures the global stability of the system (3.101). If

$$a_j = 1/C_j, \quad b_j = v_j/R_j - I_j, \quad f_j(v_j) = u_j,$$

and constant $m_{ij} = m_{ji} = T_{ij}$, the system (3.101) reduces to the Hopfield circuit model (3.100).

The Hecht–Nielsen counter-propagation network is governed by the activation equation:

$$\dot{v}_j = -Av_j + (B - v_j)I_j - v_j \sum_{k \neq j} I_k, \quad (j = 1, \dots, N),$$

where A, B are positive constants and I_j are input values for each processing unit.

Provided some simple conditions are satisfied, namely, say symmetry of weights $w_{ij} = w_{ji}$, non-negativity of activations v_j and monotonicity of transfer functions f_j , the system (1) is globally asymptotically stable (in the sense of Liapunov energy functions). The fixed-points (stable states) of the system correspond to the fundamental memories to be stored, so it works as content-addressable memory (AM). The initial state of the system (3.99) lies inside the basin of attraction of its fixed-points, so that its initial state is related to appropriate memory vector. Various variations on this basic model are reported in the literature [Hay94, Kos92], and more general form of the vector-field can be given, preserving the above stability conditions. Here is a two-step generalization.

First, all of N neurons of equation (3.99) belong to the same layer, so the M -layer generalized network is naturally governed by tensorial ‘field’-equation:

$$C_j^L \dot{v}_j^L = I_j^L - \frac{v_j^L}{R_j^L} + w_{ij}^L f_i^L(v_i^L), \quad (i, j = 1, \dots, N, \quad L = 1, \dots, M), \quad (3.102)$$

where, in the pattern-recognition formulation, js denote the pixels and Ls correspond to the clusters.

The basic two-layer example of the tensorial system (3.102) is Kosko’s adaptive bidirectional associative memory (ABAM) system, which represents a two-layer generalization of the Cohen–Grossberg system (3.101):

$$\dot{v}_j = -a_j(v_j) [b_j(v_j) - f_k(v_k) m_{jk}], \quad (j = 1, \dots, N), \quad (3.103)$$

$$\dot{u}_k = -a_k(u_k) [b_k(u_k) - f_j(u_j) m_{jk}], \quad (k = 1, \dots, N). \quad (3.104)$$

Kosko has proved the global stability for the ABAM system (3.103) in case of signal Hebbian and competitive learning [Kos92]. He has also proved the global stability for its stochastic generalization random adaptive bidirectional associative memory (RABAM) system:

$$\begin{aligned} \dot{v}_j &= -a_j(v_j) [b_j(v_j) - f_k(v_k) m_{jk}] + n_j, & (j = 1, \dots, N), \\ \dot{u}_k &= -a_k(u_k) [b_k(u_k) - f_j(u_j) m_{jk}] + n_k, & (k = 1, \dots, N). \end{aligned}$$

where $n_{j,k}$ represents an additive zero-mean Gaussian white noise.

Second, the fuzzified model of the system (3.102) can be get by replacing the inputs I_j^L with the fuzzy membership functions Ψ_j^L for each node (pixel) j in each layer (cluster) L :

$$C_j^L \dot{v}_j^L = \Psi_j^L - \frac{v_j^L}{R_j^L} + w_{ij}^L f_i^L(v_i^L), \quad (i, j = 1, \dots, N, \quad L = 1, \dots, M),$$

Other possibilities for the fuzzification include fuzzy parameters (C_j^L and R_j^L) and fuzzy initial conditions ($v_j^L(0)$). Even activation functions $f_j^L(\cdot)$ could be subject to the fuzzification process.

Learning Dynamics of Common Graded-Response Networks

A continuous unsupervised learning paradigm, i.e., continuous update law for the third-order, mixed, synaptic tensor w_{ij}^L , can be proposed in two forms (generalized from [Kos92]):

Hebbian Learning Scheme

The generalized *Hebbian learning* scheme [Heb49] is defined by

$$\dot{w}_{ij}^L = -w_{ij}^L + \Phi_{ij}^L(v_j^L, u_j^L, w_{ij}^L), \quad (i, j = 1, \dots, N, \quad L = 1, \dots, M),$$

where three terms from the left to the right denote respectively a new-update value, an old value and innovation of the synaptic tensor w_{ij}^L . In this case the nonlinear functions Φ_{ij}^L are usually defined by one of the following learning models:

(i) signal Hebbian learning, with innovation:

$$\Phi_{ij}^L = f_j^L(v_j^L) f_i^L(u_j^L);$$

(ii) differential Hebbian learning, with innovation:

$$\Phi_{ij}^L = f_j^L(v_j^L) f_i^L(u_j^L) + \dot{f}_j^L(v_j^L) \dot{f}_i^L(u_j^L),$$

where overdots denote ‘signal velocities’;

(iii) random signal Hebbian learning, with innovation:

$$\Phi_{ij}^L = f_j^L(v_j^L) f_i^L(u_j^L) + n_{ij}^L,$$

where n_{ij}^L denotes an additive, tensorial, zero-mean, Gaussian white-noise process independent of the main ‘innovation signal’; and

(iv) random differential signal Hebbian learning, with innovation:

$$\Phi_{ij}^L = f_j^L(v_j^L) f_i^L(u_j^L) + \dot{f}_j^L(v_j^L) \dot{f}_i^L(u_j^L) + n_{ij}^L.$$

Tensorial competitive learning scheme

Tensorial *competitive learning* scheme has four particular update laws:

(i) deterministic competitive learning:

$$\dot{w}_{ij}^L = f_i^L(u_j^L)[f_j^L(v_j^L) - w_{ij}^L];$$

(ii) differential competitive learning:

$$\dot{w}_{ij}^L = \dot{f}_i^L(u_j^L)[f_j^L(v_j^L) - w_{ij}^L];$$

(iii) random competitive learning, and

$$\dot{w}_{ij}^L = f_i^L(u_j^L)[f_j^L(v_j^L) - w_{ij}^L] + n_{ij}^L;$$

(iv) random differential competitive learning:

$$\dot{w}_{ij}^L = \dot{f}_i^L(u_j^L)[f_j^L(v_j^L) - w_{ij}^L] + n_{ij}^L;$$

General Neurodynamical Stability*Lyapunov stability*

A dynamical system $\dot{x}_j = F_j(x_j)$, $j = 1, \dots, N$ is stable if some Lyapunov function L decreases along the system's trajectories: $\dot{L} \leq 0$. A system is asymptotically stable if it strictly decreases along the trajectories: $\dot{L} < 0$. In a stable equilibrium the trajectory may hover arbitrarily close to the equilibrium point without reaching it. In an asymptotically stable system the state trajectory reaches the equilibrium, and in general reaches it exponentially fast. Lyapunov functions are usually chosen in a quadratic form, representing (negative) potential $L = -\sum_{j=1}^N x_j^2$ or kinetic energy $L = -\sum_{j=1}^N \dot{x}_j^2$ of the system. By the chain rule, $\dot{L} = \frac{\partial L}{\partial x_j} \dot{x}_j$. In particular, gradient systems $\dot{x}_j = -\frac{\partial L}{\partial x_j}$ are stable.

For neural-networks purposes Lyapunov functions need only decrease and be bounded [Kos92].

Convergence versus stability

In a single neural network both neurons and synapses change in time. Therefore, we have three dynamical systems [Kos92]:

- (i) the neural dynamical system $\dot{x}_j$,
- (ii) the synaptic dynamical system $\dot{w}_{jk}$, and
- (iii) the joint neuronal-synaptic dynamical system $(\dot{x}_j, \dot{w}_{jk})$.

Equilibrium is steady state (for fixed-point attractors).

Stability is neuronal equilibrium: $\dot{x}_j = 0$.

Convergence is synaptic equilibrium: $\dot{w}_{jk} = 0$.

Then global stability is joint neuronal-synaptic steady state:

both $\dot{x}_j = 0$ and $\dot{w}_{jk} = 0$ hold.

Neurons fluctuate faster than synapses. In feedback neural networks this dynamical asymmetry creates the stability-convergence dilemma: learning tends to destroy the neural patterns being learned.

GBAM Neurodynamical Classifier

System Architecture

Mathematically, the GBAM is a tensor field system (q, p, W) defined on a manifold M called the GBAM manifold. The system (q, p, W) includes two nonlinearly coupled (yet non-chaotic and stable) subsystems (see Figure 3.2): (i) activation (q, p) -dynamics, where q and p represent neuronal 1D tensor-fields, and (ii) self-organized learning W -dynamics, where W is a symmetric synaptic 2D tensor-field.

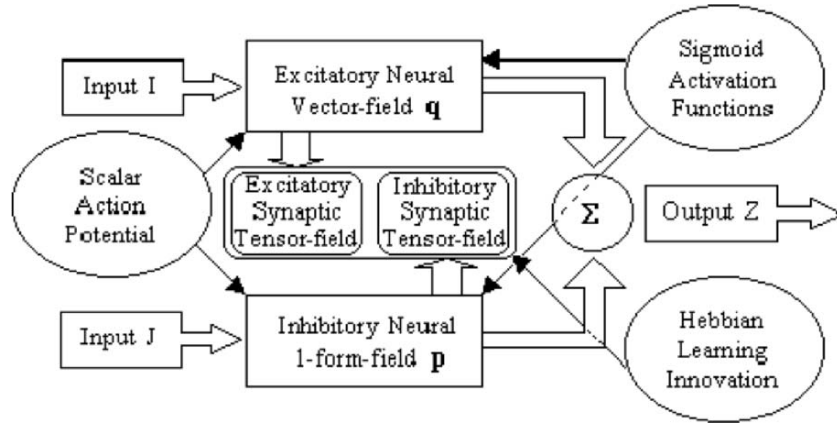


Fig. 3.2. Diagram of the GBAM neurodynamical system.

GBAM Activation Dynamics

The GBAM-manifold M can be viewed as a Banach space with a C^∞ -smooth structure on it, so that in each local chart U open in M , an n D smooth coordinate system U_α exists.

GBAM-activation (q, p) -dynamics, is defined as a system of two coupled, first-order oscillator tensor-fields, dual to each other, in a local Banach chart U_α , $\alpha = 1, \dots, n$ on M :

1. An excitatory neural vector-field $q^\alpha = q^\alpha(t) : M \rightarrow TM$, being a cross-section of the tangent bundle TM ; and
2. An inhibitory neural 1-form $p_\alpha = p_\alpha(t) : M \rightarrow T^*M$, being a cross-section of the cotangent bundle T^*M .

To start with conservative linear (q, p) -system, we postulate the GBAM scalar activation-potential V to be a negative bilinear form:

$$V = -\frac{1}{2}\omega_{\alpha\beta}q^\alpha q^\beta - \frac{1}{2}\omega^{\alpha\beta}p_\alpha p_\beta + q^\alpha p_\alpha, \quad (\alpha, \beta = 1, \dots, n), \quad (3.105)$$

where n is the number of neurons in each neural field, while $\omega_{\alpha\beta}$ and $\omega^{\alpha\beta}$ represent respectively inhibitory-covariant and excitatory-contravariant components of the symmetric (with zero-trace) coupling GBAM synaptic tensor W .

The Lyapunov-stable, conservative, linear $(q, \mathbf{p})$ -dynamics is given as a bidirectional (excitatory-inhibitory) gradient system:

$$\dot{q}^\alpha = -\frac{\partial V}{\partial p_\alpha} = \omega^{\alpha\beta} p_\beta - q^\alpha, \quad \dot{p}_\alpha = -\frac{\partial V}{\partial q^\alpha} = \omega_{\alpha\beta} q^\beta - p_\alpha. \quad (3.106)$$

As W is a symmetric and zero-trace synaptic coupling tensor, the conservative linear dynamics (3.106) is equivalent to the rule that the state of each neuron (in both excitatory and inhibitory neural fields) is changed in time if and only if the scalar action potential V , defined by relation (3.105), is lowered. Therefore, the scalar action potential V is a monotonically non-increasing Lyapunov function $\dot{V} \leq 0$ for the conservative linear (q, p) -dynamics (3.106), which converges to a local minimum or ground state of V .

Applying the inputs I^α and J_α , we get the non-conservative linear (q, p) -system equations:

$$\dot{q}^\alpha = I^\alpha + \omega^{\alpha\beta} p_\beta - q^\alpha, \quad \dot{p}_\alpha = J_\alpha + \omega_{\alpha\beta} q^\beta - p_\alpha. \quad (3.107)$$

Further, applying the sigmoid GBAM activation functions $S_\alpha(\cdot)$ and $S^\alpha(\cdot)$ to the synaptic product-terms, we get the non-conservative nonlinear (q, p) -system equations, which generalize the transient RC-circuit neurodynamical model:

$$\dot{q}^\alpha = I^\alpha + \omega^{\alpha\beta} S_\beta(p_\beta) - q^\alpha, \quad \dot{p}_\alpha = J_\alpha + \omega_{\alpha\beta} S^\beta(q^\beta) - p_\alpha. \quad (3.108)$$

The equations in (3.108) represent a 2-input system that can be applied e.g., to classification of two-feature data. The generalization to an N -input system working in a ND *feature-space* is given by

$$\dot{q}_e^\alpha = I_e^\alpha + \omega_e^{\alpha\beta} S_\beta(p_\beta) - q_e^\alpha, \quad \dot{p}_\alpha^o = I_\alpha^o + \omega_{\alpha\beta}^o S^\beta(q^\beta) - p_\alpha^o, \quad (3.109)$$

where $e(= 2, 4, \dots, N)$ and $o(= 1, 3, \dots, N-1)$ denote respectively even and odd partitions of the total sample of N features.

The GBAM model (3.109) gives a generalization of four well-known recurrent NN models:

1. Continuous Hopfield amplifier-circuit model [Hop84]

$$C_j \dot{v}_j = I_j - \frac{v_j}{R_j} + T_{ij} u_i, \quad (i, j = 1, \dots, N),$$

where $v_j = v_j(t)$ represent the activation potentials in the j th processing unit, C_j and R_j denote input capacitances and leakage resistances, $u_i = f_i[v_j(t)]$ are output functions from processing elements, and $T_{ij} = w_{ij}$ is the inverse of the resistors connection-matrix; and the functions f_i are sigmoidal.

2. Cohen–Grossberg general ART–system [CG83],

$$\dot{v}_j = -a_j(v_j)[b_j(v_j) - f_k(v_k)m_{jk}], \quad (j = 1, \dots, N),$$

with proved asymptotical stability.

3. Hecht–Nielsen counter–propagation network [Hec87],

$$\dot{v}_j = -Av_j + (B - v_j)I_j - v_jI_k,$$

where A, B are positive constants and I_j are input values for each processing unit.

4. Kosko’s BAM (ABAM and RABAM) bidirectional models [Kos92]

$$\begin{aligned} \dot{v}_j &= -a_j(v_j)[b_j(v_j) - f_k(v_k)m_{jk}], \\ \dot{u}_k &= -a_k(u_k)[b_k(u_k) - f_j(u_j)m_{jk}], \end{aligned}$$

which is globally stable for the cases of signal and random–signal Hebbian learning.

GBAM Self–Organized Learning Dynamics

The continuous (and at least C^1 –differentiable) unsupervised update law for the coupling synaptic GBAM tensor–field W can be viewed both as an inhibitory–covariant Hebbian learning scheme, generalized from [Kos92]:

$$\dot{\omega}_{\alpha\beta} = -\omega_{\alpha\beta} + \Phi_{\alpha\beta}(q^\alpha, p_\alpha), \quad (\alpha, \beta = 1, \dots, n), \quad (3.110)$$

and, as an excitatory–contravariant Hebbian learning scheme:

$$\dot{\omega}^{\alpha\beta} = -\omega^{\alpha\beta} + \Phi^{\alpha\beta}(q^\alpha, p_\alpha), \quad (3.111)$$

where the three terms from the left to the right denote respectively the new–update value, the old value and the innovation of the synaptic tensor W . In this case the nonlinear (usually sigmoid) innovation functions $\Phi_{\alpha\beta}$ and $\Phi^{\alpha\beta}$ are defined by one of following four Hebbian models:

Signal Hebbian learning, with innovation in both variance–forms:

$$\begin{aligned} \Phi_{\alpha\beta} &= S_\alpha(q^\alpha) S_\beta(p_\beta), \\ \Phi^{\alpha\beta} &= S^\alpha(q^\alpha) S^\beta(p_\beta); \end{aligned} \quad (3.112)$$

Differential Hebbian learning, with innovation in both variance–forms:

$$\begin{aligned} \Phi_{\alpha\beta} &= S_\alpha(q^\alpha) S_\beta(p_\beta) + \dot{S}_\alpha(q^\alpha) \dot{S}_\beta(p_\beta), \\ \Phi^{\alpha\beta} &= S^\alpha(q^\alpha) S^\beta(p_\beta) + \dot{S}^\alpha(q^\alpha) \dot{S}^\beta(p_\beta), \end{aligned} \quad (3.113)$$

where $\dot{S}$ –terms denote the so–called ‘signal velocities’ (for details see [Kos92]).

Random signal Hebbian learning, with innovation in both variance–forms:

$$\begin{aligned}\Phi_{\alpha\beta} &= S_\alpha(q^\alpha) S_\beta(p_\beta) + n_{\alpha\beta}, \\ \Phi^{\alpha\beta} &= S^\alpha(q^\alpha) S^\beta(p_\beta) + n^{\alpha\beta},\end{aligned}\quad (3.114)$$

where $n_{\alpha\beta} = \{n_{\alpha\beta}(t)\}$, $n^{\alpha\beta} = \{n^{\alpha\beta}(t)\}$ respectively denote covariant and contravariant additive, zero-mean, Gaussian white-noise processes independent of the main innovation signal; and

Random differential signal Hebbian learning, with innovation in both variance-forms:

$$\begin{aligned}\Phi_{\alpha\beta} &= S_\alpha(q^\alpha) S_\beta(p_\beta) + \dot{S}_\alpha(q^\alpha) \dot{S}_\beta(p_\beta) + n_{\alpha\beta}, \\ \Phi^{\alpha\beta} &= S^\alpha(q^\alpha) S^\beta(p_\beta) + \dot{S}^\alpha(q^\alpha) \dot{S}^\beta(p_\beta) + n^{\alpha\beta}.\end{aligned}\quad (3.115)$$

Total GBAM (q, p, W)—neurodynamics and biological interpretation

Total GBAM tensorial neurodynamics is defined as a union of the neural oscillatory activation (q, p)-dynamics (3.109) and the synaptic learning W -dynamics (3.115), namely

$$\begin{aligned}\dot{q}_e^\alpha &= I_e^\alpha + \omega_e^{\alpha\beta} S_\beta(p_\beta) - q_e^\alpha, \\ \dot{p}_\alpha^o &= I_\alpha^o + \omega_{\alpha\beta}^o S^\beta(q^\beta) - p_\alpha^o, \\ \dot{\omega}_e^{\alpha\beta} &= -\omega_e^{\alpha\beta} + \Phi_{e\beta}^{\alpha\beta}(q^\alpha, p_\alpha), \\ \dot{\omega}_{\alpha\beta}^o &= -\omega_{\alpha\beta}^o + \Phi_{\alpha\beta}^o(q^\alpha, p_\alpha),\end{aligned}\quad (\alpha, \beta = 1, \dots, n), \quad (3.116)$$

where the tensorial innovation Φ -functions are given by one of Hebbian models (3.112–3.115), $\alpha (= 1, \dots, n)$ is the number of continuous-time (or, graded-response) neurons in each neural-activation field, $e (= 2, 4, \dots, N)$ and $o (= 1, 3, \dots, N-1)$ denote respectively even and odd partitions of the total sample of N features.

Artificial neural networks are generally inspired by biological neural systems, but in fact, some important features of biological systems are not present in most artificial neural networks. In particular, unidirectional neural networks, which include all associative neural networks except the BAM model introduced by [Kos92], do not resemble oscillatory biological neural systems. GBAM is a generalization of Kosko's ABAM and RABAM neural systems and inherits their oscillatory (excitatory/inhibitory) neuro-synaptic behavior. Such oscillatory behavior is a basic characteristic of a number of biological systems. Examples of similar oscillatory neural ensembles in the human nervous system are:

- Motoneurons and Renshaw interneurons in the spinal cord;
- Pyramidal and basket cells in the hippocampus;
- Mitral and granule cells in the olfactory bulb;
- Pyramidal cells and thalamic inter-neurons in cortico-thalamic system;
- Interacting excitatory and inhibitory populations of neurons found in the cerebellum, olfactory cortex, and neocortex, all representing the basic

mechanisms for the generation of oscillating (EEG-monitored) activity in the brain.

Therefore, GBAM can be considered as a model for any of above-mentioned oscillatory biological neural systems.

Self-Organizing Lie-Derivative Neuro-Classifier

A *Lie-derivative neuro-classifier* is a self-organized, associative-memory machine, represented by oscillatory (excitatory/inhibitory) tensor-field-system (x, v, ω) on the Banach manifold M . It consists of continual neural activation (x, y) -dynamics and self-organizing synaptic learning ω -dynamics.

The continual activation (x, y) -dynamics, is defined as a system of two coupled, first-order oscillator tensor-fields, dual to each other, in a local Banach chart U_α , $(\alpha = 1, \dots, n)$ on M :

- 1) an excitatory neural vector-field $x^i = x^i(t) : M \rightarrow TM$, representing a cross-section of the tangent bundle TM ; and
- 2) an inhibitory neural one-form $y_i = y_i(t) : M \rightarrow T^*M$, representing a cross-section of the cotangent bundle T^*M .

The self-organized learning ω -dynamics is performed on a second-order symmetrical synaptic tensor-field $\omega = \omega(t)$, given by its covariant components $\omega_{ij} = \omega_{ij}(t)$ and its contravariant components $\omega^{ij} = \omega^{ij}(t)$, where $i, j = 1, \dots, n$.

Starting with the Lyapunov-stable, negative scalar neural action potential:

$$U = -\frac{1}{2}(\omega_{ij}x^ix^j + \omega^{ij}y_iy_j), \quad (i, j = 1, \dots, n),$$

the (x, y) -dynamics is given in two versions, which are compared and contrasted:

- (1) the **Lie-linear** neurodynamics with first-order Lie derivatives

$$\dot{x}^i = J^i + L_X U, \quad \dot{y}_i = J_i + L_Y U,$$

and

- (2) the **Lie-quadratic** neurodynamics with both first and second-order Lie derivatives

$$\dot{x}^i = J^i + L_X U + L_X L_X U, \quad \dot{y}_i = J_i + L_Y U + L_Y L_Y U,$$

where $X = S_i(x^i)$, $Y = S_i(y_i)$, S_i represent sigmoid activation functions, while $L_X L_X, L_Y L_Y : F(M) \rightarrow F(M)$ denote the *second-order (iterated) Lie derivatives*.

Self-organized learning ω -dynamics is presented in the form of differential Hebbian learning scheme in both covariant and contravariant forms

$$\begin{aligned} \dot{\omega}_{ij} &= -\omega_{ij} + S_i(x^i)S_j(y_j) + \dot{S}_i(x^i)\dot{S}_j(y_j), & \text{and} \\ \dot{\omega}^{ij} &= -\omega^{ij} + S_i(x^i)S_j(y_j) + \dot{S}_i(x^i)\dot{S}_j(y_j), & (i, j = 1, \dots, n), \end{aligned}$$

respectively.

Self-Organizing Lie-Poisson Neuro-Classifier

A *Lie-Poisson neuro-classifier* is a tensor-field-system $\{\mu\} = (q, p, \omega)$ on a Poisson manifold $(\mathbf{g}^*, \{F, H(\mu)\}_\pm(\mu))$. Like a GBAM neuro-classifier, it consists of continual activation (q, p) -dynamics and self-organized learning ω -dynamics. In this case, both dynamics are defined by “neural activation form” of the abstract Lie-Poisson evolution equation

$$\dot{F} = \{S(F), H(\mu)\}, \quad (3.117)$$

where $S(\cdot) = \tanh(\cdot)$ denotes the sigmoid activation function. A Hamiltonian function $H(\mu)$, representing the total network energy, is given in the form

$$H(\mu) = \frac{1}{2}\omega_{ij}\delta^{ij} + \frac{1}{2}\omega^{ij}\delta_{ij}, \quad (i, j = 1, \dots, n),$$

where δ^{ij} and δ_{ij} are Kronecker tensors, while $\omega^{ij} = \omega^{ij}(q^i)$ and $\omega_{ij} = \omega_{ij}(p_i)$ correspond to the contravariant and covariant components of the functional-coupling synaptic tensor $\omega = \omega(q, p)$, defined respectively by

$$\omega^{ij} = \varepsilon q^i q^j, \quad \omega_{ij} = \tau p_i p_j,$$

with random coefficients ε and τ .

Activation (q, p) -dynamics are given by

$$\dot{q}^i = I^i + \{S(q^i), H(\mu)\}, \quad \dot{p}_i = J_i + \{S(p_i), H(\mu)\},$$

where I^i and J_i represent the two input features.

Two types of self-organized learning ω -dynamics are presented and compared:

Lie-Poisson learning dynamics, in which synaptic update law is given by inhibitory-covariant and excitatory-contravariant form of equation (3.117):

$$\dot{\omega}_{ij} = \{S(\omega_{ij}), H(\mu)\}, \quad \dot{\omega}^{ij} = \{S(\omega^{ij}), H(\mu)\},$$

respectively.

Differential Hebbian learning (see [Kos92] for details), in both inhibitory-covariant and excitatory-contravariant learning form:

$$\dot{\omega}_{ij} = -\omega_{ij} + \Phi_{ij}(q^i, p_i), \quad \dot{\omega}^{ij} = -\omega^{ij} + \Phi^{ij}(q^i, p_i),$$

with innovations defined in both variance-forms as:

$$\Phi_{ij} = S_i(q^i)S_j(p_j) + \dot{S}_i(q^i)\dot{S}_j(p_j), \quad \Phi^{ij} = S^i(q^i)S^j(p_j) + \dot{S}^i(q^i)\dot{S}^j(p_j).$$

3.2.3 Darwinian Oscillatory Neural Net

Recall that classical *embedding field equation* (see [II06a]) of the nonlinear oscillatory neural network $\mathcal{N}$ can be presented in the covariant form

$$\dot{x}^i = -D_j^i x^i + T_j^i g^i(x^i) + S^i, \quad (3.118)$$

where $x^i = x^i(t)$ represents the mean soma potential of the i th neuron; $g^i(x^i(t))$ is a smooth function that represents the short-term average of the firing rate of the i th neuron as a function of its potential; S^i is the constant external input to the network; D_j^i is diagonal tensor and its elements $a_i^i > 0$ represent the time constant of the rate of change for the i th neuron's potential; T_j^i is the mixed second order tensor-field whose elements represent synaptic efficiency of the j th neuron's potential transmitted to the i th neuron.

For the proposed model $\mathcal{N}$ sufficient conditions for existence and asymptotic stability of the network's equilibria were derived using perturbation theory (see [BDJ89]) and reduced to a set of piecewise linear inequality relations that can be solved by feedforward binary network. The storage capacity of the network was analyzed using geometric analysis based on the function g that models the firing rate of the system.

Each neuron in the network $\mathcal{N}$ is regarded as a computing process and the interconnections between neurons as *virtual communication links* embedded in the hypercube topology. Associated with each neuron is a decision algorithm that determines its next state. This algorithm is akin to the cellular automata approach in that each neuron uses only local information to reach its decision.

According to the system (3.118) a computer model of a frequency-coded neural network $\mathcal{N}^f$ for self-organization, learning, and coupling to simulated pulse-coded sensors was analyzed in [BDJ89]. A single i th neuron in this system $\mathcal{N}$, neglecting environmental influences, behaves according to the difference scheme $\mathcal{N}^d$, given by

$$\Delta x^i = b x^i (1 - y^i) - (x^i - P) y^i, \quad \Delta y^i = -v y^i + g(x^i + T_j^i y^j),$$

where $v > 0$ is the rate of decay of the action potential, $0 < b < v$ is the charging rate of the neuron, P is the value to which each neuron charges, and $g(x^i)$ is the Heaviside unit step function translated to a firing threshold $G > 0$. The term $(1 - y_i)$ acts to suppress the charging term when the node fires. Allowing the soma charging to compete with the axon firing, and choosing P to be zero, we may pass to a continuous system representing a forced oscillator $\mathcal{N}^c$

$$\dot{x}^i = (b - y^i) x^i, \quad \dot{y}^i = -v y^i + g(x^i + T_j^i y^j). \quad (3.119)$$

This departure from the usual first-order membrane equations results in interesting dynamic behavior of the model. If the synaptic weights T_j^i are identically zero, these equations specify a nonlinear, free oscillator with a typical (noncircular) limit cycle [GOY87].

Each oscillator in this frequency-coded model $\mathcal{N}^f$ is coupled to many identical oscillators [BDJ89]. The coupling is through the tensor-field T_j^i of synaptic weights. It has the following properties:

1. It is nonsymmetric and biased in a preferred direction within the network structure;
2. It has a dynamic evolution in time due to interaction with the environment; and
3. It represents a nonlinear feedback mechanism.

The field of oscillators, a threshold-feedback mechanism for focusing attention, the synaptic field for recording an internal representation of the external environment, and the external environment itself make up the essential components of the global system $\mathcal{N}$ [BDJ89].

Since each neuron in $\mathcal{N}$ is an oscillator with external feedback and an external forcing function (input from other neurons of the environment), phase changes in the system represent an important mode of information processing. The relative phase of an information loop (local attractor in the node firing dynamics) may be changed by incoming information having a higher frequency than the network average. The attention mechanism of the threshold control system ensures that the new information will be processed by the network until the stimulus drops back below the network average.

The possibility of chaotic behavior in a dynamical neural system $\mathcal{N}$ makes *a priori* knowledge of system parameters (sensor firing rates, node refractory period, noise level of membrane potential) difficult if not impossible [GOY87]. If the system parameters are thought of as genes in a genetic space $\mathcal{N}^p$ of possible synthetic creatures, then the methods of nature may be used [DK87]. Darwinian variation and selection (with the natural competitive environment playing the selector role) can be abstracted and simplified [KS86] by merging the concepts of variation and mutation into a single random parameter change each generation, followed by a ‘lifetime’ during which the synthetic system is allowed to organize itself according to environmental constraints. The part of natural selection is played by an algorithm measuring the fitness of each genome instantiation.

Random mutations are made in the parameter set $\mathcal{N}^p$ at the rate of one for each instantiation, and a change is accepted once each generation [DK87, KS86]. Which gene to vary, the amount of variation, and its sign are chosen randomly and limited by a control gene. If the variation is a success (i.e., if the daughter performs equal to or better than the mother), the daughter becomes a parent and another mutation of the same amount in the same direction is made in the parameter that failed. If both directions (increase and decrease) result in failure, a local maximum along a specific axis has probably been found, and another parameter is chosen at random for variation. Successful as well as neutral changes are thus accumulated, resulting in a process of behavioral selection and improvement over time.

A mechanism for evolving the genome necessitates appropriate mapping of the genes onto the parameter set $\mathcal{N}^p$ as well as allowing for the appearance of new genes and the disappearance of old genes when they are no longer functional [DK87, KS86]. Not only will an individual synthetic creature evolve, but an optimal region in the genetic landscape will also be found. Successful evolution of the entire quasi species is seen as determining a bundle of trajectories in the embedding space. Any particular trajectory in the bundle represents an evolutionary history of the genome up to the end point.

The most natural application of such dynamical neural-net system $\mathcal{N}$ can be found in a skeletal muscle system $\mathcal{M}$ response to *efferent functional stimulation* (EFS) $\mathcal{F}$ and its motor control system $\mathcal{F}^{-1}$ (see [II06a]).

3.2.4 Oscillatory Phase Synchronization in Neurodynamics

Phase Synchronization in Coupled Chaotic Oscillators

Over the past decade or so, *synchronization in chaotic oscillators* [FY83, PC90] has received much attention because of its fundamental importance in nonlinear dynamics and potential applications to laser dynamics [DBO01], electronic circuits [KYR98], chemical and biological systems [ESH98], and secure communications [KP95]. Synchronization in chaotic oscillators is characterized by the loss of exponential instability in the transverse direction through interaction. In coupled chaotic oscillators, it is known, various types of synchronization are possible to observe, among which are *complete synchronization* (CS) [FY83, PC90], *phase synchronization* (PS) [RPK96, ROH98], *lag synchronization* (LS) [RPK97] and *generalized synchronization* (GS) [KP96].

One of the noteworthy synchronization phenomena in this regard is PS which is defined by the phase locking between nonidentical chaotic oscillators whose amplitudes remain chaotic and uncorrelated with each other: $|\theta_1 - \theta_2| \leq \text{const}$. Since the first observation of PS in mutually coupled chaotic oscillators [RPK96], there have been extensive studies in theory [ROH98] and experiments [DBO01]. The most interesting recent development in this regard is the report that the inter-dependence between physiological systems is represented by PS and *temporary phase-locking* (TPL) states, e.g., (a) *human heart beat and respiration* [SRK98], (b) a certain brain area and the tremor activity [TRW98, RGL99]. Application of the concept of PS in these areas sheds light on the analysis of non-stationary bivariate data coming from biological systems which was thought to be impossible in the conventional statistical approach. And this calls new attention to the PS phenomenon [KK00, KLR03].

Accordingly, it is quite important to elucidate a detailed transition route to PS in consideration of the recent observation of a TPL state in biological systems. What is known at present is that TPL[ROH98] transits to PS and then transits to LS as the coupling strength increases. On the other hand, it is noticeable that the phenomenon from non-synchronization to PS have hardly

been studied, in contrast to the wide observations of the TPL states in the biological systems.

In this section, following [KK00, KLR03], we study the characteristics of TPL states observed in the regime from non-synchronization to PS in coupled chaotic oscillators. We report that there exists a special locking regime in which a TPL state shows maximal periodicity, which phenomenon we would call *periodic phase synchronization* (PPS). We show this PPS state leads to local negativeness in one of the vanishing Lyapunov exponents, taking the measure by which we can identify the maximal periodicity in a TPL state. We present a qualitative explanation of the phenomenon with a nonuniform oscillator model in the presence of noise.

We consider here the unidirectionally coupled non-identical Roessler oscillators for first example:

$$\begin{aligned}\dot{x}_1 &= -\omega_1 y_1 - z_1, & \dot{y}_1 &= \omega_1 x_1 + 0.15 y_1, & \dot{z}_1 &= 0.2 + z_1(x_1 - 10.0), \\ \dot{x}_2 &= -\omega_2 y_2 - z_2, & \dot{y}_2 &= \omega_2 x_2 + 0.165 y_2 + \epsilon(y_1 - y_2), \\ \dot{z}_2 &= 0.2 + z_2(x_2 - 10.0),\end{aligned}\tag{3.120}$$

where the subscripts imply the oscillators 1 and 2, respectively, $\omega_{1,2}$ ($= 1.0 \pm 0.015$) is the overall frequency of each oscillator, and ϵ is the coupling strength. It is known that PS appears in the regime $\epsilon \geq \epsilon_c$ and that 2π phase jumps arise when $\epsilon < \epsilon_c$. Lyapunov exponents play an essential role in the investigation of the transition phenomenon with coupled chaotic oscillators and as generally understood that PS transition is closely related to the transition to the negative value in one of the vanishing Lyapunov exponents [PC90].

A vanishing Lyapunov exponent corresponds to a phase variable of an oscillator and it exhibits the neutrality of an oscillator in the phase direction. Accordingly, the local negativeness of an exponent indicates this neutrality is locally broken [RPK96]. It is important to define an appropriate phase variable in order to study the TPL state more thoroughly. In this regard, several methods have been proposed methods of using linear interpolation at a Poincaré section [RPK96], phase-space projection [RPK96, ROH98], tracing of the center of rotation in phase-space [YL97], Hilbert transformation [RPK96], or wavelet transformation [KK00, KLR03]. Among these we take the method of phase-space projection onto the $x_1 - y_1$ and $x_2 - y_2$ planes with the geometrical relation

$$\theta_{1,2} = \arctan(y_{1,2}/x_{1,2}),$$

and get *phase difference* $\varphi = \theta_1 - \theta_2$.

The system of coupled oscillators is said to be in a TPL state (or laminar state) when $\langle \varphi \rangle < \Lambda_c$ where $\langle \dots \rangle$ is the running average over appropriate short time scale and Λ_c is the cutoff value to define a TPL state. The locking length of the TPL state, τ , is defined by time interval between two adjacent peaks of $\langle \varphi \rangle$.

In order to study the characteristics of the locking length τ , we introduce a measure [KK00, KLR03]: $P(\epsilon) = \sqrt{\text{var}(\tau)}/\langle\tau\rangle$, which is the ratio between the average value of time lengths of TPL states and their standard deviation. In terminology of stochastic resonance, it can be interpreted as noise-to-signal ratio [PK97, Jun93]. The measure would be minimized where the periodicity is maximized in TPL states.

To validate the argument, we explain the phenomenon in simplified dynamics. From (3.120), we get the equation of motion in terms of phase difference:

$$\dot{\varphi} = \Delta\omega + A(\theta_1, \theta_2, \epsilon) \sin \varphi + \xi(\theta_1, \theta_2, \epsilon), \quad \text{where} \quad (3.121)$$

$$\begin{aligned} A(\theta_1, \theta_2, \epsilon) &= (\epsilon + 0.15) \cos(\theta_1 + \theta_2) - \frac{\epsilon}{2} \left(\frac{R_1}{R_2} \right), \\ \xi(\theta_1, \theta_2, \epsilon) &= \frac{\epsilon}{2} \frac{R_1}{R_2} \sin(\theta_1 + \theta_2) + \frac{z_1}{R_1} \sin(\theta_1) - \frac{z_2}{R_2} \sin(\theta_2) \\ &\quad + (\epsilon + 0.015) \cos(\theta_2) \sin(\theta_2). \end{aligned}$$

$$\text{Here,} \quad \Delta\omega = \omega_1 - \omega_2, \quad R_{1,2} = \sqrt{x_{1,2}^2 + y_{1,2}^2}.$$

And from (3.121) we get the simplified equation to describe the phase dynamics:

$$\dot{\varphi} = \Delta\omega + \langle A \rangle \sin(\varphi) + \xi,$$

where $\langle A \rangle$ is the time average of $A(\theta_1, \theta_2, \epsilon)$. This is a nonuniform oscillator in the presence of noise where ξ plays a role of effective noise [Str94] and the value of $\langle A \rangle$ controls the width of bottleneck (i.e., non-uniformity of the flow). If the bottleneck is wide enough, (i.e., faraway from the saddle-node bifurcation point: $\Delta\omega \gg -\langle A \rangle$), the effective noise hardly contributes to the phase dynamics of the system. So the passage time is wholly governed by the width of the bottleneck as follows:

$$\langle\tau\rangle \sim 1/\sqrt{\Delta\omega^2 - \langle A \rangle^2} \sim 1/\sqrt{\Delta\omega^2 - \epsilon^2/4},$$

which is a slowly increasing function of ϵ . In this region while the standard deviation of TPL states is nearly constant (because the widely opened bottlenecks periodically appears and those lead to small standard deviation), the average value of locking length of TPL states is relatively short and the ratio between them is still large.

On the contrary as the bottleneck becomes narrower (i.e., near the saddle-node bifurcation point: $\Delta\omega \geq -\langle A \rangle$) the effective noise begins to perturb the process of bottleneck passage and regular TPL states develop into intermittent ones [ROH98, KK00]. It makes the standard deviation increase very rapidly and this trend overpowers that of the average value of locking lengths of the TPL states. Thus we understand that the competition between width of bottleneck and amplitude of effective noise produces the crossover at the minimum point of $P(\epsilon)$ which shows the maximal periodicity of TPL states.

Rosenblum *et al.* firstly observed the dip in mutually coupled chaotic oscillators [RPK96]. However the origin and the dynamical characteristics of the dip have been left unclarified. We argue that the dip observed in mutually coupled chaotic oscillators has the same origin as observed above in unidirectionally coupled systems.

Common apprehension is that near the border of synchronization the phase difference in coupled regular oscillators is periodic [RPK96] whereas in coupled chaotic oscillators it is irregular [ROH98]. On the contrary, we report that the special locking regime exhibiting the maximal periodicity of a TPL state also exists in the case of coupled chaotic oscillators. In general, the phase difference of coupled chaotic oscillators is described by the 1D Langevin equation,

$$\dot{\varphi} = F(\varphi) + \xi,$$

where ξ is the effective noise with finite amplitude. The investigation with regard to PS transition is the study of scaling of the laminar length around the virtual fixed-point φ^* where $F(\varphi^*) = 0$ [KK00, KT01] and PS transition is established when

$$\left| \int_{\varphi}^{\varphi^*} F(\varphi) d\varphi \right| > \max |\xi|.$$

Consequently, the crossover region, from which the value of P grows exponentially, exists because intermittent series of TPL states with longer locking length τ appears as PS transition is nearer. Eventually it leads to an exponential growth of the standard deviation of the locking length. Thus we argue that PPS is the generic phenomenon mostly observed in coupled chaotic oscillators prior to PS transition.

In conclusion, analyzing the dynamic behaviors in coupled chaotic oscillators with slight parameter mismatch we have completed the whole transition route to PS. We find that there exists a special locking regime called PPS in which a TPL state shows maximal periodicity and that the periodicity leads to local negativeness in one of the vanishing Lyapunov exponents. We have also made a qualitative description of this phenomenon with the nonuniform oscillator model in the presence of noise. Investigating the characteristics of TPL states between non-synchronization and PS, we have clarified the transition route before PS. Since PPS appears in the intermediate regime between non-synchronization and PS, we expect that the concept of PPS can be used as a tool for analyzing weak inter-dependences, i.e., those not strong enough to develop to PS, between non-stationary bivariate data coming from biological systems, for instance [KK00, KLR03]. Moreover PPS could be a possible mechanism of the chaos regularization phenomenon [Har92, Rul01] observed in neurobiological experiments.

Oscillatory Phase Neurodynamics

In coupled oscillatory neuronal systems, under suitable conditions, the original dynamics can be reduced theoretically to a simpler phase dynamics. The

state of the i th neuronal oscillatory system can be then characterized by a single phase variable φ_i representing the timing of the neuronal firings. The typical dynamics of *oscillator neural networks* are described by the *Kuramoto model* [Kur84, HI97, Str00], consisting of N equally weighted, all-to-all, phase-coupled limit-cycle oscillators, where each oscillator has its own natural frequency ω_i drawn from a prescribed distribution function:

$$\dot{\varphi}_i = \omega_i + \frac{K}{N} J_{ij} \sin(\varphi_j - \varphi_i + \beta_{ij}). \quad (3.122)$$

Here, J_{ij} and β_{ij} are parameters representing the effect of the interaction, while $K \geq 0$ is the coupling strength. For simplicity, we assume that all natural frequencies ω_i are equal to some fixed value ω_0 . We can then eliminate ω_0 by applying the transformation $\varphi_i \rightarrow \varphi_i + \omega_0 t$. Using the complex representation $W_i = \exp(i\varphi_i)$ and $C_{ij} = J_{ij} \exp(i\beta_{ij})$ in (3.122), it is easily found that all neurons relax toward their stable equilibrium states, in which the relation $W_i = h_i/|h_i|$ ($h_i = C_{ij}W_j$) is satisfied. Following this line of reasoning, as a synchronous update version of the oscillator neural network we can consider the alternative discrete form [AN99],

$$W_i(t+1) = \frac{h_i(t)}{|h_i(t)|}, \quad h_i(t) = C_{ij}W_j(t). \quad (3.123)$$

Now we will attempt to construct an extended model of the oscillator neural networks to retrieve sparsely coded phase patterns. In equation (3.123), the complex quantity h_i can be regarded as the local field produced by all other neurons. We should remark that the phase of this field, h_i , determines the timing of the i th neuron at the next time step, while the amplitude $|h_i|$ has no effect on the retrieval dynamics (3.123). It seems that the amplitude can be thought of as the strength of the local field with regard to emitting spikes. Pursuing this idea, as a natural extension of the original model we stipulate that the system does not fire and stays in the resting state if the amplitude is smaller than a certain value. Therefore, we consider a network of N oscillators whose dynamics are governed by

$$W_i(t+1) = f(|h_i(t)|) \frac{h_i(t)}{|h_i(t)|}, \quad h_i(t) = C_{ij}W_j(t). \quad (3.124)$$

We assume that $f(x) = \Theta(x - H)$, where the real variable H is a threshold parameter and $\Theta(x)$ is the unit step function; $\Theta(x) = 1$ for $x \geq 0$ and 0 otherwise. Therefore, the amplitude $|W_i^t|$ assumes a value of either 1 or 0, representing the state of the i th neuron as firing or non-firing. Consequently, the neuron can emit spikes when the amplitude of the local field $h_i(t)$ is greater than the threshold parameter H .

Now, let us define a set of P patterns to be memorized as $\xi_i^\mu = A_i^\mu \exp(i\theta_i^\mu)$ ($\mu = 1, 2, \dots, P$), where θ_i^μ and A_i^μ represent the phase and the amplitude of the i th neuron in the μ th pattern, respectively. For simplicity, we assume that

the θ_i^μ are chosen at random from a uniform distribution between 0 and 2π . The amplitudes A_i^μ are chosen independently with the probability distribution

$$P(A_i^\mu) = a\delta(A_i^\mu - 1) + (1 - a)\delta(A_i^\mu),$$

where a is the mean activity level in the patterns. Note that, if $H = 0$ and $a = 1$, this model reduces to (3.123).

For the synaptic efficacies, to realize the function of the associative memory, we adopt the *generalized Hebbian rule* in the form

$$C_{ij} = \frac{1}{aN} \xi_i^\mu \tilde{\xi}_j^\mu, \quad (3.125)$$

where $\tilde{\xi}_j^\mu$ denotes the complex conjugate of ξ_j^μ . The overlap $M_\mu(t)$ between the state of the system and the pattern μ at time t is given by

$$M_\mu(t) = m_\mu(t) e^{i\varphi_\mu(t)} = \frac{1}{aN} \tilde{\xi}_j^\mu W_j(t), \quad (3.126)$$

In practice, the rotational symmetry forces us to measure the correlation of the system with the pattern μ in terms of the amplitude component $m_\mu(t) = |M_\mu(t)|$.

Let us consider the situation in which the network is recalling the pattern ξ_1^1 ; that is, $m_1(t) = m(t) \sim O(1)$ and $m_\mu(t) \sim O(1/\sqrt{N})$ ($\mu \neq 1$). The local field $h_i(t)$ in (3.124) can then be separated as

$$h_i(t) = C_{ij} W_j(t) = m_t e^{i\varphi_1(t)} \xi_i^1 + z_i(t), \quad (3.127)$$

where $z_i(t)$ is defined by

$$z_i(t) = \frac{1}{aN} \xi_i^\mu \tilde{\xi}_j^\mu W_j(t). \quad (3.128)$$

The first term in (3.127) acts to recall the pattern, while the second term can be regarded as the noise arising from the other learned patterns. The essential point in this analysis is the treatment of the second term as *complex Gaussian noise* characterized by

$$\langle z_i(t) \rangle = 0, \quad \langle |z_i(t)|^2 \rangle = 2\sigma(t)^2. \quad (3.129)$$

We also assume that $\varphi_1(t)$ remains a constant, that is, $\varphi_1(t) = \varphi_0$. By applying the method of statistical neurodynamics to this model under the above assumptions [AN99], we can study the retrieval properties analytically. As a result of such analysis we have found that the retrieval process can be characterized by some macroscopic order parameters, such as $m(t)$ and $\sigma(t)$.

From (3.126), we find that the overlap at time $t + 1$ is given by

$$m(t + 1) = \left\langle \left\langle f(|m(t) + z(t)|) \frac{m(t) + z(t)}{|m(t) + z(t)|} \right\rangle \right\rangle, \quad (3.130)$$

where $\langle\langle \dots \rangle\rangle$ represents an average over the complex Gaussian $z(t)$ with mean 0 and variance $2\sigma(t)^2$. For the noise $z(t+1)$, in the limit $N \rightarrow \infty$ we get [AN99]

$$z_i(t+1) \sim \frac{1}{aN} \xi_i^\mu \tilde{\xi}_j^\mu f(|h_{j,\mu}(t)|) \frac{h_{j,\mu}(t)}{|h_{j,\mu}(t)|} + z_i(t) \left(\frac{f'(|h_{j,\mu}(t)|)}{2} + \frac{f(|h_{j,\mu}(t)|)}{2|h_{j,\mu}(t)|} \right), \quad (3.131)$$

where $h_{j,\mu}(t) = 1/aN \xi_j^\nu \tilde{\xi}_k^\nu W_k(t)$.

Kuramoto Synchronization Model

The microscopic individual level dynamics of the Kuramoto model (3.122) is easily visualized by imagining oscillators as points running around on the unit circle. Due to rotational symmetry, the average frequency $\Omega = \sum_{i=1}^N \omega_i / N$ can be set to 0 without loss of generality; this corresponds to observing dynamics in the co-rotating frame at frequency Ω .

The governing equation (3.122) for the i th oscillator phase angle φ_i can be simplified to

$$\dot{\varphi}_i = \omega_i + \frac{K}{N} \sum_{j=1}^N \sin(\varphi_j - \varphi_i), \quad 1 \leq i \leq N. \quad (3.132)$$

It is known that as K is increased from 0 above some critical value K_c , more and more oscillators start to get synchronized (or phase-locked) until all the oscillators get fully synchronized at another critical value of K_{tp} . In the choice of $\Omega = 0$, the fully synchronized state corresponds to an exact steady state of the ‘detailed’, fine-scale problem in the co-rotating frame.

Such synchronization dynamics can be conveniently summarized by considering the fraction of the synchronized (phase-locked) oscillators, and conventionally described by a *complex-valued order parameter* [Kur84, Str00], $re^{i\psi} = \frac{1}{N} \sum e^{i\varphi_j}$, where the radius r measures the phase coherence, and ψ is the average phase angle.

Transition from Full to Partial Synchronization

Following [MK05], here we restate certain facts about the nature of the second transition mentioned above, a transition between the full and the partial synchronization regime at $K = K_{tp}$, in the direction of decreasing K .

A fully synchronized state in the continuum limit corresponds to the solution to the mean-field type alternate form of equation (3.132),

$$\dot{\varphi}_i = \Omega = \omega_i + rK \sum_{j=1}^N \sin(\psi - \varphi_i), \quad (3.133)$$

where Ω is the common angular velocity of the fully synchronized oscillators (which is set to 0 in our case). Equation (3.133) can be further rewritten as

$$\frac{\Omega - \omega_i}{rK} = \sum_{i=1}^N \sin(\psi - \varphi_i), \quad (3.134)$$

where the absolute value of the r.h.s is bounded by unity.

As K approaches K_{tp} from above, the l.h.s for the ‘extreme’ oscillator (the oscillator in a particular family that has the maximum value of $|\Omega - \omega_i|$) first exceeds unity, and a real-valued solution to (3.134) ceases to exist. Different random draws of ω_i ’s from $g(\omega)$ for a finite number of oscillators result in slightly different values of K_{tp} . K_{tp} appears to follow the Gumbel type extreme distribution function [KN00], just as the maximum values of $|\Omega - \omega_i|$ do:

$$p(K_{tp}) = \sigma^{-1} e^{-(K_{tp}-\mu)/\sigma} \exp[-e^{-(K_{tp}-\mu)/\sigma}],$$

where σ and μ are parameters.

Lyapunov Chaotic Synchronization

The notion of *conditional Lyapunov exponents* was introduced by Pecora and Carroll in their study of synchronization of chaotic systems. First, in [PC91], they generalized the idea of driving a stable system to the situation when the drive signal is chaotic. This led to the concept of conditional Lyapunov exponents and also generalized the usual criteria of the linear stability theorem. They showed that driving with chaotic signals can be done in a robust fashion, rather insensitive to changes in system parameters. The calculation of the stability criteria led naturally to an estimate for the convergence of the driven system to its stable state. The authors focussed on a homogeneous driving situation that led to the construction of synchronized chaotic subsystems. They applied these ideas to the Lorenz and Rössler systems, as well as to an electronic circuit and its numerical model. Later, in [PC98], they showed that many coupled oscillator array configurations considered in the literature could be put into a simple form so that determining the stability of the synchronous state could be done by a master stability function, which could be tailored to one’s choice of stability requirement. This solved, once and for all, the problem of synchronous stability for any linear coupling of that oscillator.

It turns out, that, like the full Lyapunov exponent, the conditional exponents are well defined ergodic invariants, which are reliable quantities to quantify the relation of a global dynamical system to its constituent parts and to characterize dynamical self-organization [Men98].

Given a dynamical system defined by a map $f : M \rightarrow M$, with $M \subset R^m$ the conditional exponents associated to the splitting $R^k \times R^{m-k}$ are the eigenvalues of the limit

$$\lim_{n \rightarrow \infty} (D_k f^{n*}(x) D_k f^n(x))^{\frac{1}{2n}},$$

where $D_k f^n$ is the $k \times k$ diagonal block of the full Jacobian.

Mendes [Men98] proved that existence of the conditional Lyapunov exponents as well-defined ergodic invariants was guaranteed under the same conditions that established the existence of the Lyapunov exponents.

Recall that for measures μ that are absolutely continuous with respect to the Lebesgue measure of M or, more generally, for measures that are smooth along unstable directions (SBR measures) Pesin's [Pes77] identity holds

$$h(\mu) = \sum_{\lambda_i > 0} \lambda_i,$$

relating *Kolmogorov-Sinai entropy* $h(\mu)$ to the sum of the Lyapunov exponents. By analogy we may define the *conditional exponent entropies* [Men98] associated to the splitting $R^k \times R^{m-k}$ as the sum of the positive conditional exponents counted with their multiplicity

$$h_k(\mu) = \sum_{\xi_i^{(k)} > 0} \xi_i^{(k)}, \quad h_{m-k}(\mu) = \sum_{\xi_i^{(m-k)} > 0} \xi_i^{(m-k)}.$$

The Kolmogorov-Sinai entropy of a dynamical system measures the rate of information production per unit time. That is, it gives the amount of randomness in the system that is not explained by the defining equations (or the minimal model [CY89]). Hence, the conditional exponent entropies may be interpreted as a measure of the randomness that would be present if the two parts $S^{(k)}$ and $S^{(m-k)}$ were uncoupled. The difference $h_k(\mu) + h_{m-k}(\mu) - h(\mu)$ represents the effect of the coupling.

Given a dynamical system S composed of N parts $\{S_k\}$ with a total of m degrees of freedom and invariant measure μ , one defines a *measure of dynamical self-organization* $I(S, \Sigma, \mu)$ as

$$I(S, \Sigma, \mu) = \sum_{k=1}^N \{h_k(\mu) + h_{m-k}(\mu) - h(\mu)\}.$$

For each system S , this quantity will depend on the partition Σ into N parts that one considers. $h_{m-k}(\mu)$ always denotes the conditional exponent entropy of the complement of the subsystem S_k . Being constructed out of ergodic invariants, $I(S, \Sigma, \mu)$ is also a well-defined ergodic invariant for the measure μ . $I(S, \Sigma, \mu)$ is formally similar to a mutual information. However, not being strictly a mutual information, in the information theory sense, $I(S, \Sigma, \mu)$ may take negative values.

3.2.5 Synchronization in Cellular Automata

Recall from Introduction that cellular automata (CA) are discrete dynamical systems that may exhibit complex space-time patterns. It has been observed

that CA may be synchronized by a stochastic coupling [BR95]. In this subsection, following [BR99], we study all CA with four, five and six neighbors. For the CA considered, we find that a synchronization threshold is reached critically and that all CA with complex non-periodic space-time patterns have a positive threshold and conversely. We present a strong relationship between the synchronization threshold and the maximum Lyapunov exponent (MLE) of CA.

Considering the following asymmetric coupling for a continuous 1D map $f(x)$

$$\begin{aligned}x(t+1) &= f[x(t)], \\ y(t+1) &= (1-p)f[y(t)] + pf[x(t)],\end{aligned}$$

with $0 \leq p \leq 1$. The function $f(x)$ depends in general on one (or more) parameter(s) a ; let us assume that, for the chosen value of a , $f(x)$ is chaotic with Lyapunov exponent λ , and that $x(0) \neq y(0)$. Then, $x(t)$ is always different from $y(t)$ for $p = 0$, while for $p = 1$ x and y synchronize in one time step. There exists a critical synchronization threshold p_c for which both trajectories $x(t)$ and $y(t)$ become indistinguishable in the long time limit and

$$p_c = 1 - \exp(-\lambda). \quad (3.135)$$

In this subsection we will develop similar relations for CA, following [BR95, BR99]. We begin with a brief review of the definition of maximum Lyapunov exponent for CA based on a linear expansion of the evolution rule. We then present a synchronization mechanism and show that the distance between two realizations goes to zero in a critical manner at p_c . The numerical experiments show a relation between p_c and the maximum Lyapunov exponent which may be understood by considering several probabilistic CA. We restrict our study to one dimensional, *totalistic Boolean CA* with four, five and six inputs, since their number is reasonably manageable and their evolution can be efficiently implemented.

A Boolean CA, F , of range r is defined as a map on the set of configurations $\{\mathbf{x}\}$ with $\mathbf{x} = (x_0, \dots, x_{N-1})$, $x_i = 0, 1$, and $i = 0, \dots, N-1$ such that

$$\mathbf{x}(t+1) = F[\mathbf{x}(t)],$$

where $t = 0, 1, \dots$. The map F is defined locally on every site i by

$$x_i(t+1) = f(\{x_i(t)\}_r),$$

where $\{x_i\}_r = (x_i, \dots, x_{i+r-1})$ is the neighborhood of range r of site i at time t , assuming periodic boundary conditions. For totalistic CA, the local function f is symmetric and depends only on s defined by

$$s(\{x_i\}_r) = \sum_{j=0}^{r-1} x_{i+j}.$$

That is $x_i(t+1) = f(s[\{x_i(t)\}_r])$. It is useful to introduce the following operations between Boolean quantities: the sum modulo two (XOR), denoted by the symbol $\oplus$, and the AND operation, which is analogous to the usual multiplication and shares the same symbol. These operations can be performed between two configurations component by component. We introduce the difference, or damage, $\mathbf{z}(\mathbf{t}) = \mathbf{x}(\mathbf{t}) \oplus \mathbf{y}(\mathbf{t})$, whose evolution is given by $\mathbf{z}(t+1) = F[\mathbf{x}(t)] \oplus F[\mathbf{y}(t)]$ and we define the norm of $\mathbf{z}$ as

$$|\mathbf{z}| = (1/N) \sum_i x_i \oplus y_i.$$

A function $f(x_i, \dots, x_j, \dots, x_{i+r})$ is sensitive to its j th argument for a given neighborhood $(\{x_i\}_r)$ if the Boolean derivative

$$\left. \frac{\partial f}{\partial x_j} \right|_{\{x_i\}_r} = f(x_i, \dots, x_j, \dots) \oplus f(x_i, \dots, x_j \oplus 1, \dots) = 1.$$

The Jacobian matrix J of F is an $N \times N$ matrix with components

$$J_{i,j}(\mathbf{x}) = \left. \frac{\partial f}{\partial x_j} \right|_{\{x_i\}_r}.$$

The matrix J is circular with zeroes everywhere except possibly on the main diagonal and the following $r-1$ upper diagonals.

It is possible to ‘Taylor expand’ a Boolean function around a given point using *Boolean derivatives*. To first order in $|\mathbf{z}|$ we have [BR99]

$$F(\mathbf{y}) = F(\mathbf{x}) \oplus J(\mathbf{x}) \odot \mathbf{z}, \quad (3.136)$$

where $\odot$ denotes the Boolean multiplication of a matrix by a vector. Compared to algebraic multiplication of a matrix by a vector, the sum and multiplication of scalars are replaced by the XOR and the AND operations respectively. Using (3.136) we may approximate the evolution of the damage configuration $\mathbf{z}$ by

$$\mathbf{z}(t+1) = J(\mathbf{x}) \odot \mathbf{z}.$$

However, $|\mathbf{z}|$ grows at most linearly with t since a damage cannot spread to more than r neighbors in one time step: a fixed site i at time $t+1$ can be damaged if at least one of its r neighbors at time t is damaged, but if more than one of the neighbors is damaged, the damage may cancel. Since

$$z_i(t+1) = \bigoplus_{j=i}^{i+r-1} J_{i,j}(\mathbf{x}) z_j,$$

$z_i(t+1) = 1$ if $J_{i,j}(\mathbf{x}) z_j = 1$ on an odd number of sites. In order to account for all possible damage spreading we choose to consider each damage independently. If, at time t , m damaged sites are present, we consider m replicas

each one with a different damaged site. On each replica, the damage evolves for one time step, without interference effects and so on.

This procedure is equivalent to choosing a vector $\boldsymbol{\xi}(0) = \mathbf{z}(0)$ that evolves in time according to

$$\boldsymbol{\xi}(t+1) = J(\mathbf{x})\boldsymbol{\xi}, \quad (3.137)$$

where now the matrix multiplication is algebraic. The components ξ_i are positive integers that count the number of ways in which the initial damage can spread to site i at time t on the ensemble of replicas.

We define the maximum Lyapunov exponent λ of the CA F by

$$\lambda(\mathbf{x}^0) = \lim_{T \rightarrow \infty} \lim_{N \rightarrow \infty} \frac{1}{T} \log \left(\frac{|\boldsymbol{\xi}^T|}{|\boldsymbol{\xi}^0|} \right),$$

where $|\boldsymbol{\xi}|$ may be taken as the Euclidean norm or as the sum of its components. The geometrical average μ of ones in the Jacobian matrix J is defined by

$$\mu(\mathbf{x}^0) = \lim_{T \rightarrow \infty} \lim_{N \rightarrow \infty} \left(\prod_{t=0}^{T-1} \frac{1}{r} \sum_{i,j} J_{i,j}(\mathbf{x}^t) \right)^{1/T}.$$

The process defined by (3.137) may be viewed as a deterministic directed bond percolation problem where a site i at depth t is wet if $\xi_i(t) > 0$. The bonds exist where the components of J are 1. A first approximation is obtained by replacing J with a random matrix whose elements are zero except on the diagonal and the $r-1$ following upper diagonals, where they are equal to one with probability μ [BR99]. There is a critical value $\mu_c(r)$ below which the bond percolation process falls into the absorbing state so that the maximum eigenvalue of the product of random matrices is zero. We can further introduce a mean field approximation to the directed bond percolation process, which exhibits discrepancies only very near to μ_c . In this case one can show that

$$\tilde{\lambda} = \ln(r\mu) \quad (3.138)$$

is an upper bound to the MLE of the product of random matrices.

We now discuss a synchronization mechanism for CA. Starting with two initial configurations chosen at random $\mathbf{x}(0)$ and $\mathbf{y}(0)$ we have [BR99]

$$\begin{aligned} \mathbf{x}(t+1) &= F[\mathbf{x}(t)] \\ \mathbf{y}(t+1) &= \overline{S^t(p)} F[\mathbf{y}(t)] \oplus S^t(p) F[\mathbf{x}(t)], \end{aligned}$$

where $S^t(p)$ is a Boolean random diagonal matrix with elements $s_i^t(p)$ that, at each time step, take the value one with probability p and zero with probability $1-p$; $\overline{S(p)} = I - S(p)$ and I is the identity matrix. On the average, y'_i will be set to the value of $x'_i = f(\{x_i\})$ on a fraction p of sites. In this way we introduced a stochastic synchronization mechanism over a deterministic process. This

stochastic mechanism can be considered as a ‘random field’ approximation of an intermittent coupling generated by a deterministic chaotic process. The evolution equation for the difference $\mathbf{z} = \mathbf{x} \oplus \mathbf{y}$ is

$$\mathbf{z}(t+1) = \overline{S(p)}\{F[\mathbf{x}(t)] \oplus F[\mathbf{y}(t)]\}. \quad (3.139)$$

The control and order parameters are p and $h(p) = \lim_{t \rightarrow \infty} \lim_{N \rightarrow \infty} |\mathbf{z}(t)|$ respectively. We say that $\mathbf{x}$, the driver, and $\mathbf{y}$, the driven system, synchronize when $h(p) = 0$. For $p = 0$ both systems evolve independently, while for $p = 1$ they synchronize in just one step; we expect then to find a synchronization threshold p_c . This behavior is shared by all the CA with complex non periodic space–time patterns. All others synchronize for $p \simeq 0$. This can be conversely expressed by saying that all CA that synchronize with a nontrivial p_c exhibit complex non periodic space–time patterns.

For totalistic linear rules, whose evolution is given by

$$f(\{x_i\}_r) = \bigoplus_{j=0}^{r-1} x_{i+j},$$

the synchronization equation (3.139) is equivalent to the dilution (with probability $1 - p$) of the rule.

Let us study the relation between p_c and λ by some random approximations. Near the synchronization threshold p_c , we may expand $\mathbf{y}$ around $\mathbf{x}$ with the help of (3.136) so that

$$\mathbf{z}(t+1) = \overline{S(p)}J(\mathbf{x})\mathbf{z}. \quad (3.140)$$

Equation (3.140) may be written as

$$z_i(t+1) = \overline{s_i(p)} [J_{i,i}(\mathbf{x})z_i \oplus \cdots \oplus J_{i,i+r-1}(\mathbf{x})z_{i+r-1}]. \quad (3.141)$$

During the time evolution of a particular CA a fixed value of μ is attained so that on average $r\mu$ derivatives inside the parenthesis on the *r.h.s.* of (3.141) are different from zero. A first approximation, model *A*, is obtained by replacing the derivatives with random variables $m_i(\mu)$ that are one with probability μ and zero with probability $(1 - \mu)$. That is

$$z_i(t+1) = \overline{s_i(p)} [m_i(\mu)z_i \oplus \cdots \oplus m_{i+r-1}(\mu)z_{i+r-1}].$$

On every site and every time step, $r\mu$ variables are chosen on average and if their sum is odd, then with probability p , $z_i(t+1) = 1$. We then look for the synchronization threshold $p_c(\mu)$ and plot it as a function of $\tilde{\lambda} = \ln(r\mu)$, given by (3.138).

Since the typical ‘complex’ CA pattern exhibit transient correlations (‘triangles’), one can model them by choosing a fixed number $k \leq r$ of derivatives equal to one. The simplest way is to take $k = r\mu$ with $r\mu$ an integer, model *B*, which . Then

$$z_i(t+1) = \overline{s_i(p)} [z_i \oplus \cdots \oplus z_{i+k-1}].$$

which is a dilution of the XOR with k inputs. This process is expected to belong to the same universality class of *directed percolation*.

We can extend this last model allowing non-integer values of $r\mu$ by

$$z_i(t+1) = \overline{s_i^t(p)} m_i^t(\mu) [z_i \oplus \cdots \oplus z_{i+k-1}]$$

which we call model C_k . In this way the average number of non zero derivatives is $k\mu$ with $0 \leq \mu \leq 1$. Now k is a free parameter, and this model can be useful to delimit the expected spread of (λ, p_c) points.

Since $s_i^t(p)$ and $m_i^t(\mu)$ are independent random variables we may write

$$z_i(t+1) = \overline{s_i^t(q)} [z_i \oplus \cdots \oplus z_{i+k-1}].$$

where $(1-q) = (1-p)\mu$. In this guise, this is model B with k inputs. The synchronization threshold is given by

$$p_c(k, \mu) = 1 - (1 - q_c(k))/\mu,$$

where $q_c(k)$ is the percolation threshold of the dilution of the XOR with k inputs. Using the approximation $\lambda = \ln(k\mu)$, one has

$$p_c(k, \lambda) = 1 - k[1 - q_c(k)] \exp(-\lambda).$$

For the totalistic one dimensional CA with $r = 4, 5, 6$ we can safely say that all CA with a positive p_c exhibit complex non periodic space-time patterns and conversely. These CA also have a positive MLE. We also showed that the synchronization of CA is a critical phenomenon similar to directed percolation.

We proposed several approximations based on a combination of ‘linearization’ of CA rules using Boolean Taylor expansions and stochasticity and showed the relation between the synchronization threshold and the MLE. In particular, model C implies a relation similar to that found for continuous maps with the addition of a percolation threshold constant.

An analogous mechanism can be applied to coupled map lattices; in this case p is the probability that a map $y_i(t)$ takes the value $x_i(t)$. One observes a synchronization transitions, but the critical value p_c is not correlated to the usual MLE. For more details, see [BR99] and references therein.

3.2.6 Kuramoto-Type Dynamics of Complex Nets

Recall that many systems in nature, such as neural nets, food webs, metabolic systems, co-authorship of papers, the worldwide web, etc. can be represented as *complex networks*, or *small-world networks* (see, e.g., [WS98, DM03]). In particular, it has been recognized that many networks have scale-free topology; the distribution of the degree obeys the power law, $P(k) \sim k^{-\gamma}$. The

study of the scale-free network now attracts the interests of many researchers in mathematics, physics, engineering and biology [Ich04].

Another important aspect of complex networks is their dynamics, describing e.g., the spreading of viruses in the Internet, change of populations in a food web, and synchronization of neurons in a brain. In particular, [Ich04] studied the synchronization of the random network of oscillators. His work follows the previous studies (see [Str00]) that showed that mean-field type synchronization, that Kuramoto observed in globally-coupled oscillators [Kur84], appeared also in the small-world networks.

Continuum Limit of the Kuramoto Network

Ichinomiya started with the standard network with N nodes, described by a variant of the *Kuramoto model* (3.122). At each node, there exists an oscillator and the phase of each oscillator θ_i is evolving according to

$$\dot{\theta}_i = \omega_i + K \sum_j a_{ij} \sin(\theta_j - \theta_i), \quad (3.142)$$

where K is the coupling constant, a_{ij} is 1 if the nodes i and j are connected, and 0 otherwise; ω_i is a random number, whose distribution is given by the function $N(\omega)$.

For the analytic study, it is convenient to use the *continuum limit equation*. We define $P(k)$ as the distribution of nodes with degree k , and $\rho(k, \omega; t, \theta)$ the density of oscillators with phase θ at time t , for given ω and k . We assume that $\rho(k, \omega; t, \theta)$ is normalized as

$$\int_0^{2\pi} \rho(k, \omega; t, \theta) d\theta = 1.$$

For simplicity, we also assume $N(\omega) = N(-\omega)$. Thus, we suppose that the collective oscillation corresponds to the stable solution, $\dot{\rho} = 0$.

Now we construct the continuum limit equation for the network of oscillators. The evolution of ρ is determined by the *continuity equation* $\partial_t \rho = -\partial_\theta(\rho v)$ (see Appendix), where v is defined by the continuum limit of the r.h.s of (3.142). Because one randomly selected edge connects to the node of degree k , frequency ω , phase θ with the probability $kP(k)N(\omega)\rho(k, \omega; t, \theta)/\int dk kP(k)$, $\rho(k, \omega; t, \theta)$ obeys the equation

$$\begin{aligned} \partial_t \rho(k, \omega; t, \theta) = & -\partial_\theta [\rho(k, \omega; t, \theta) (\omega \\ & + \frac{Kk \int d\omega' \int dk' \int d\theta' N(\omega') P(k') k' \rho(k', \omega'; t, \theta') \sin(\theta - \theta')}{\int dk' P(k') k'})]. \end{aligned}$$

The mean-field solution of this equation was studied by [Ich04].

Path-Integral Approach to Complex Nets

Recently, [Ich05] introduced the *path-integral* (see subsection 4.4.6 above) *approach* in studying the dynamics of complex networks. He considered the stochastic generalization of the Kuramoto network (3.142), given by

$$\dot{x}_i = f_i(x_i) + \sum_{j=1}^N a_{ij} g(x_i, x_j) + \xi_i(t), \quad (3.143)$$

where $f_i = f_i(x_i)$ and $g_{ij} = g(x_i, x_j)$ are functions of network activations x_i , $\xi_i(t)$ is a *random force* that satisfies $\langle \xi_i(t) \rangle = 0$, $\langle \xi_i(t) \xi_j(t') \rangle = \delta_{ij} \delta(t - t') \sigma^2$. He assumed $x_i = x_{i,0}$ at $t = 0$. In order to discuss the dynamics of this system, he introduced the so-called *Matrin-Siggia-Rose* (MSR) *generating functional* Z (see [Dom78]) given by

$$Z[\{l_{ik}\}, \{\bar{l}_{ik}\}] = \left(\frac{1}{\pi}\right)^{NN_t} \left\langle \int \prod_{i=1}^N \prod_{k=0}^{N_t} dx_{ik} d\bar{x}_{ik} e^{-S} \exp(l_{ik} x_{ik} + \bar{l}_{ik} \bar{x}_{ik}) J \right\rangle,$$

where the *action* S is given by

$$S = \sum_{ik} \left[\frac{\sigma^2 \Delta t}{2} \bar{x}_{ik}^2 + i \bar{x}_{ik} \{x_{ik} - x_{i,k-1} - \Delta t(f_i(x_{i,k-1}) + \sum_j a_{ij} g(x_{i,k-1}, x_{j,k-1}))\} \right],$$

and $\langle \dots \rangle$ represents the average over the ensemble of networks. J is the functional Jacobian term,

$$J = \exp \left(-\frac{\Delta t}{2} \sum_{ijk} \frac{\partial(f_i(x_{ik}) + a_{ij} g(x_{ik}, x_{jk}))}{\partial x_{ik}} \right).$$

Ichinomiya considered such a form of the network model (3.143) in which

$$a_{ij} = \begin{cases} 1 & \text{with probability } p_{ij}, \\ 0 & \text{with probability } 1 - p_{ij}. \end{cases}$$

Note that p_{ij} can be a function of variables such as i or j . For example, in the 1D chain model, p_{ij} is 1 if $|i - j| = 1$, else it is 0. The average over all networks can be expressed as

$$\begin{aligned} & \left\langle \exp \left[\sum_{ik} i \Delta t \bar{x}_{ik} \sum_j a_{ij} g(x_{i,k-1}, x_{j,k-1}) \right] \right\rangle \\ &= \prod_{ij} \left[p_{ij} \exp \left\{ \sum_k i \Delta t \bar{x}_{ik} g(x_{i,k-1}, x_{j,k-1}) \right\} + 1 - p_{ij} \right], \end{aligned}$$

so we get

$$\langle e^{-S} \rangle = \exp(-S_0) \prod_{ij} \left[p_{ij} \exp \left\{ \sum_k i \Delta t \bar{x}_{ik} g(x_{i,k-1}, x_{j,k-1}) \right\} + 1 - p_{ij} \right],$$

where $S_0 = \sum_{ik} \frac{\sigma^2 \Delta t}{2} \bar{x}_{ik}^2 + i \bar{x}_{ik} \{x_{ik} - x_{i,k-1} - \Delta t f_i(x_{i,k-1})\}.$

This expression can be applied to the dynamics of any complex network model. [Ich05] applied this model to analysis of the *Kuramoto transition in random sparse networks*.

3.2.7 Oscillator AM with Scattered Natural Frequencies

Historically, there are two important studies regarding the non-equilibrium systems with large degrees of freedoms that cannot be treated by conventional equilibrium statistical mechanics, since we can not define the Lyapunov function with ‘bottoms’. Kuramoto [Kur84] theoretically analyzed the mutual entrainment of uniformly coupled oscillators with scattered natural frequencies (*Kuramoto model*). His model corresponds to a mean field model of a ferromagnet in equilibrium statistical mechanics. Kuramoto utilized the spilit of statistical mechanics, namely, a notion of macroscopic *order-parameters*, to investigate his non-equilibrium system with large degrees of freedoms. Daido [Dai92] numerically analyzed the quasi-entrainment of randomly coupled oscillators with scattered natural frequencies. His model corresponds to the *Sherrington–Kirkpatrick model* of a spin glass [SK75] in equilibrium statistical mechanics. A mean-field theory should be developed for non-equilibrium random systems *with* frustration that follows the history of mean field theories in the equilibrium statistical mechanics. However, this kind of non-equilibrium random system has not yet been theoretically analyzed (see [Dai87] for the random system without frustration). This is a famous ‘open problem’ in physics [HH95].

On the other hand, the Lyapunov function can not be defined in spin-like systems with the non-monotonic reaction function either. In the self-consistent signal-to-noise analysis (SCSNA) [SF92], the notion of a macroscopic order parameter is also introduced to analyze these frustrated spin-like systems, which cannot be treated by conventional equilibrium statistical mechanics. Note that the results of applying the SCSNA to simple random spin systems [SF92, Oka95], i.e., the Sherrington–Kirkpatrick model and the Hopfield model [Hop82], coincide with those of the replica theories [KBH91].

In this subsection, following [AKO99], we present a new theoretical framework for an oscillator associative memory model with scattered natural frequencies in memory retrieval states. This system can be considered as a typical example of non-equilibrium random systems with large degrees of freedoms. The present model makes a bridge between the SCSNA and the Kuramoto

model. Using the same procedure, we can easily treat a glass oscillator system [Dai92]. Our theory is reduced to the Kuramoto model in the finite loading case. When all oscillators have uniform natural frequencies, our theory coincides with the previously proposed models [Coo89] in the equilibrium statistical mechanics for an XY spin system.

The mutual entrainment is an important notion of a non-equilibrium system with large degrees of freedoms. In uniformly coupled oscillators, there is a unique stable state, the ferromagnetic phase in the phase space. On the other hand, in frustrated systems, there are many stable states in the phase space. We need to elucidate the properties of the mutual entrainment in each stable phase (ferromagnetic phase and glass phase). The presented model describes a phenomenon of the mutual entrainment in the ferromagnetic phase (memory retrieval). Thus, we numerically study a degree of the mutual entrainment in the glass phase (spurious memory retrieval). Almost all oscillators synchronize under memory retrieval, but desynchronize under spurious memory retrieval when setting optimal parameters [AKO99]. Thus, it is possible to determine whether the recalling process is successful or not using information about the synchrony/asynchrony.

In general, when the coupling is sufficiently weak, the high-dimensional dynamics of a coupled oscillator system can be reduced to the phase equation [Kur84, Erm96]. Let us consider the following simplified model,

$$\dot{\varphi}_i = \omega_i + \sum_{j=1}^N J_{ij} \sin(\varphi_j - \varphi_i + \beta_{ij}),$$

where N is the total number of oscillators, φ_i is the phase of the i -th oscillator, and ω_i is the natural frequency assumed to be randomly distributed over the whole population with a density denoted by the symmetric distribution $g(\omega)$, i.e., $g(\omega) = g(-\omega)$. Note that the average of ω_i may be set to zero without loss of generality. The model presented below can be easily extended to treat the system with other any distribution $g(\omega)$. J_{ij} and β_{ij} denote an amplitude of a synaptic weight and a synaptic delay, respectively. In order to investigate the nature of frustrated non-equilibrium systems, we have selected the following generalized Hebb learning rule to determine J_{ij} and β_{ij} ,

$$C_{ij} = J_{ij} \exp(i\beta_{ij}) = \frac{1}{N} \xi_i^\mu \bar{\xi}_j^\mu, \quad \xi_i^\mu = \exp(i\theta_i^\mu),$$

as their typical example. $\{\theta_i^\mu\}_{i=1, \dots, N, \mu=1, \dots, p}$ are the phase patterns to be stored in the network and are assigned to random numbers with a uniform probability in $[0, 2\pi]$. Here, we define a parameter α (loading rate) such that $\alpha = p/N$. In the equilibrium limit of this model, that is, $g(\omega) = \delta(\omega)$, the storage capacity given by $\alpha_c = 0.038$ [Coo89].

We put $s_i = \exp(i\phi_i)$ for the sake of simplicity. The order parameter m^μ , which measures the overlap between system state s_i and embedded pattern ξ^μ , is defined as

$$m^\mu = \frac{1}{N} \bar{\xi}_j^\mu s_j.$$

We get order parameter equations of the present system by applying the following manipulations: (i) Assuming a self-consistent local field for each of oscillators, a distribution of s_i under $g(\omega_i)$ is formally derived by the Bergamot model; (ii) we estimate the contribution of randomness, that is, the uncondensed patterns in the present case by the SCSNA, and determine the local field in a self-consistent manner; (iii) The order parameter equations are obtained using the self-consistent local field. We assume $m^1 = O(1)$ and $m^\mu = O(1/\sqrt{N})$ for $\mu > 1$ (uncondensed patters). Then, the following two dimensional equations for the order parameters are obtained,

$$m = \langle\langle X(x_1, x_2, \xi) \rangle\rangle_{x_1, x_2, \xi}, \quad (3.144)$$

$$U = \langle\langle F_1(x_1, x_2, \xi) \rangle\rangle_{x_1, x_2, \xi}, \quad (3.145)$$

where $\langle\langle \cdots \rangle\rangle_{x_1, x_2, \xi}$ is taken to mean the Gaussian average over x_1, x_2 and condensed pattern ξ^1 , $\langle\langle \cdots \rangle\rangle_{x_1, x_2, \xi} = \langle \int \int D x_1 D x_2 \cdots \rangle_\xi$. The pattern superscripts 1 of m are omitted for brevity. The self-consistent mean field $\tilde{h}$, X , F_1 and the Gaussian measure $D x_1 D x_2$ are expressed as [AKO99],

$$\begin{aligned} D x_1 D x_2 &= \frac{d x_1 d x_2}{2\pi\rho^2} \exp\left(-\frac{x_1^2 + x_2^2}{2\rho^2}\right), \\ \rho^2 &= \frac{\alpha}{2(1-U)^2}, \quad \tilde{h} = \xi m + x_1 + i x_2, \\ X(x_1, x_2, \xi) &= \tilde{h} \int_{-1}^1 dx g(|\tilde{h}|x) \sqrt{1-x^2}, \\ F_1(x_1, x_2, \xi) &= \int_{-1}^1 dx \left(g(|\tilde{h}|x) + \frac{|\tilde{h}|}{2} x g'(|\tilde{h}|x) \right) \sqrt{1-x^2}. \end{aligned}$$

Here, U corresponds to the susceptibility, which measures the sensitivity to external fields. A distribution of resultant frequencies $\bar{\omega}$ in the memory retrieval state, which is denoted as $p(\bar{\omega})$, becomes

$$\begin{aligned} p(\bar{\omega}) &= r \delta(\bar{\omega}) + \int D x_1 D x_2 \frac{g\left(\bar{\omega} \sqrt{1 + \frac{|h|^2}{\bar{\omega}^2}}\right)}{\sqrt{1 + \frac{|h|^2}{\bar{\omega}^2}}}, \\ r &= \int D x_1 D x_2 |\tilde{h}| \int_{-1}^1 dx g(|\tilde{h}|x), \end{aligned} \quad (3.146)$$

where r measures the ratio between the number of synchronous oscillators and the system size N . We now consider the relationships between the present model and the previously proposed models. For the equilibrium limit, $g(x) = \delta(x)$, we get [AKO99]

$$X = \frac{\tilde{h}}{|\tilde{h}|}, \quad F_1 = \frac{1}{2|\tilde{h}|}, \quad p(\bar{\omega}) = \delta(\bar{\omega}),$$

which coincide with the replica theory [Coo89] and the SCSNA [SKO04]. On the other hand, regarding the uniform-system limit, $\alpha = 0$, the presented model reproduces the Kuramoto model as

$$\begin{aligned} m &= m \int_{-1}^1 dx g(|m|x) \sqrt{1-x^2}, \\ p(\bar{\omega}) &= r\delta(\bar{\omega}) + \frac{g\left(\bar{\omega}\sqrt{1+\frac{|m|^2}{\bar{\omega}^2}}\right)}{\sqrt{1+\frac{|m|^2}{\bar{\omega}^2}}}, \\ r &= |m| \int_{-1}^1 dx g(|m|x). \end{aligned}$$

Now, if C_{ij} is assigned to random numbers with Gaussian in complex plane,

$$\text{Re}[C_{ij}] \sim \mathcal{N}(1/N, \alpha/2N), \quad \text{Im}[C_{ij}] \sim \mathcal{N}(0, \alpha/2N),$$

the order parameter equation consists with (3.144) under constraint $U = 0$. We can also treat Daido's glass oscillator system (with real number interaction) [Dai92] using the same procedure.

Next, we examine the distributions of the resultant frequencies $\bar{\omega}_i$ over the whole population of oscillators in the memory retrieval state and the spurious memory state. Here, the resultant frequencies $\bar{\omega}_i$ were calculated by using the long time average of $d\phi_i/dt$. There exists a region σ that satisfies the two conditions: all oscillators mutually synchronize in memory retrieval, and oscillators de-synchronize in spurious memory retrieval. Thus, it is possible to determine whether the recalling process is successful or not only using the information about the synchrony/asynchrony, when proper σ is given.

The phase description proposed here can be considered as a minimum model of neural networks based on oscillatory activities that is mathematically solvable. Chaos neural networks yield rich phenomena as discussed here, but can not be easily analyzed, except with simulations. Since the present analysis corresponds to the replica symmetric approximation, we have noted that it should be extended to the replica symmetry breaking in order to properly treat the spurious states (spin glass states).

In the field of neuroscience, a growing number of researchers have been interested in the synchrony of oscillatory neural activities because physiological evidence of their existence has been obtained in the visual cortex of a cat [GKE89]. Much experimental and theoretical research exists regarding the functional role of synchronization. One of the more interesting hypotheses is called *synchronized population coding*, which was proposed by Phillips and Singer [PS97]. However, its validity is highly controversial. If we consider

information processing in brain systems, the solvable toy model presented in this subsection may be a good candidate for showing the validity of a synchronized population coding in the brain, and we believe the present analysis may strongly influence debate on the functional role of synchrony [AKO99].

3.2.8 AM Neural Nets with Self-Control

An important property required in efficient neural network modeling is an autonomous functioning independent from, e.g., external constraints or control mechanisms. For fixed-point retrieval by an attractor associative memory model this requirement is mainly expressed by the robustness of its learning and retrieval capabilities against external noise, against malfunctioning of some of the connections and so on. Indeed, a model which embodies this robustness is able to perform as a *content-addressable memory* having large basins of attraction for the memorized patterns. Intuitively one can imagine that these basins of attraction become smaller when the storage capacity gets larger. This might occur, e.g., in sparsely coded models (see [Oka96] and references cited therein). Therefore the necessity of a control of the activity of the neurons has been emphasized such that the latter stays the same as the activity of the memorized patterns during the recall process. In particular, for binary patterns with a strong bias some external constraints were proposed on the dynamics in order to realize this [AGS87, Ama89, BDS89, SSP96].

An interesting question is then whether the recall process can be optimized without imposing such external constraints, keeping the simplicity of the (given) architecture of the network model. To this end a self-control mechanism has been proposed in the dynamics of binary neuron models through the introduction of a time-dependent threshold in the transfer function. This threshold is determined in function of both the cross-talk noise and the activity of the memorized patterns in the network and adapts itself in the course of the time evolution [DB98].

In this section, following [BDA00], we address the influence of a macroscopic time-dependent threshold on the retrieval dynamics of *attractor AM-models* with ternary neurons.

Networks with Ternary Neurons

Let us consider a network with N ternary neurons. At a discrete time step t the neurons $\sigma_{i,t} \in \{0, \pm 1\}$, $i = 1, \dots, N$ are updated synchronously according to the rule [BDA00]

$$\sigma_{i,t+1} = F_{\theta_t}(h_{i,t}), \quad h_{i,t} = J_{ij}\sigma_{j,t}, \quad (3.147)$$

where $h_{i,t}$ is usually called the ‘local field’ [HKP93] of neuron i at time t . In general, the transfer function F_{θ_t} can be a monotonic function with θ_t a time-dependent threshold parameter. Later on it will be chosen as

$$F_{\theta_t}(x) = \begin{cases} \text{sign}(x) & \text{if } |x| > \theta_t \\ 0 & \text{if } |x| < \theta_t. \end{cases} \quad (3.148)$$

In the sequel, for theoretical simplicity in the methods used, the number of neurons N will be taken to be sufficiently large.

The synaptic weights J_{ij} are determined as a function of the memorized patterns $\xi_i^\mu \in \{0, \pm 1\}$, ($i = 1, \dots, N$, $\mu = 1, \dots, p$), by the following learning algorithm

$$J_{ij} = \frac{C_{ij}}{Ca} N J_{ij}^H, \quad J_{ij}^H = \frac{1}{N} \xi_i^\mu \xi_j^\mu. \quad (3.149)$$

In this learning rule the J_{ij}^H are the standard *Hebbian weights* [Heb49, HKP93] with the ternary patterns ξ_i^μ taken to be *independent identically distributed random variables* (IIDRV) chosen according to the probability distribution

$$p(\xi_i^\mu) = a\delta(|\xi_i^\mu|^2 - 1) + (1 - a)\delta(\xi_i^\mu). \quad (3.150)$$

Here $a = \langle |\xi_i^\mu|^2 \rangle$ is the activity of the memorized patterns which is taken to be the same for all μ and which is given by the limit $N \rightarrow \infty$ of

$$a_N^\mu \equiv \frac{1}{N} \sum_i |\xi_i^\mu|^2. \quad (3.151)$$

The brackets $\langle \dots \rangle$ denote the average over the memorized patterns. The latter are unbiased and uncorrelated, i.e.,

$$\langle \xi_i^\mu \rangle = 0, \quad \langle \xi_i^\mu \xi_i^\nu \rangle = 0.$$

To get the J_{ij} themselves the Hebbian weights J_{ij}^H are multiplied with the $C_{ij} \in \{0, 1\}$ which are chosen to be IIDRV with probability [BDA00]

$$\Pr(C_{ij} = 1) = C/N, \quad \Pr(C_{ij} = C_{ji}) = (C/N)^2, \quad C/N \ll 1, C > 0.$$

This introduces the so-called extremely diluted asymmetric architecture with C measuring the average connectivity of the network [DGZ87].

Note that the couplings (3.149) are of infinite range (each neuron interacts with infinitely many others) such that our model allows a so-called *mean-field theory approximation*. This essentially means that we focus on the dynamics of a single neuron while replacing all the other neurons by an average background local field. In other words, no fluctuations of the other neurons are taken into account, not even in response to changing the state of the chosen neuron. In our case this approximation becomes exact because, crudely speaking, $h_{i,t}$ is the sum of very many terms and a central limit theorem can be applied [HKP93].

It is standard knowledge by now that synchronous *mean-field theory dynamics* can be solved exactly for these diluted architectures (e.g., [BSV94]). Hence, the big advantage is that this will allow us to determine the precise effects from self-control in an exact way.

In order to measure the quality of the recall process one usually introduces the Hamming distance between the microscopic state of the network model and the μ -th memorized pattern, defined as

$$d_t^\mu \equiv \frac{1}{N} \sum_i |\xi_i^\mu - \sigma_{i,t}|^2 = a_N^\mu - 2a_N^\mu m_{N,t}^\mu + q_{N,t}.$$

This relation naturally leads to the definition of retrieval overlap between the μ th pattern and the network state

$$m_{N,t}^\mu \equiv \frac{1}{Na_N^\mu} \xi_i^\mu \sigma_{i,t}, \quad (3.152)$$

and the activity of the neurons, called neural activity

$$q_{N,t} \equiv \frac{1}{N} \sum_i |\sigma_{i,t}|^2. \quad (3.153)$$

The $m_{N,t}^\mu$ are normalized parameters within the interval $[-1, 1]$ which attain the maximal value 1 whenever the model succeeds in a perfect recall, i.e., $\sigma_{i,t} = \xi_i^\mu$ for all i .

Alternatively, the precision of the recall process can be measured by the *performance parameter* [Rie90, SWB97]

$$P_t^\mu \equiv \frac{1}{N} \sum_i \delta_{\xi_i^\mu, \sigma_{i,t}} \quad (3.154)$$

which counts the relative number of correctly recalled bits. For subsequent manipulation, it is expedient to note that δ_{ξ_i, σ_i} can be expressed as a linear combination of terms $\xi_i^k \sigma_i^l$ with $k, l \leq 2$

$$\delta_{\sigma, \xi} = 1 - \sigma^2 - \xi^2 + \frac{1}{2} \sigma \xi + \frac{3}{2} \sigma^2 \xi^2. \quad (3.155)$$

Once the parameters (3.152) and (3.153) are known, both these measures for retrieval can be calculated via the dynamics (3.147). Here we remark that for associative memory models with neurons having more than three states these measures for the retrieval quality can be defined in the same way. Then, technically speaking, the performance parameter (3.154)–(3.155) will contain higher-order powers of $\sigma \xi$.

Recently an information theoretic concept, the mutual information [Sha48, Bla90] has been introduced in the study of the quality of recall of some theoretical and practical network models (see [DB98, ST98] and references therein). For sparsely coded networks in particular it turns out that this concept is very useful and, in fact, to be preferred [DB98] above the Hamming distance.

At this point we note that it turns out to be important to introduce the following quantity appearing in the performance

$$n_{N,t}^\mu \equiv \frac{1}{Na_N^\mu} \sigma_{i,t}^2 (\xi_i^\mu)^2. \quad (3.156)$$

We call this quantity the *activity-overlap* since it determines the overlap between the active neurons and the active parts of a memorized pattern. Although it does not play any independent role in the time evolution of the associative memory model defined here it appears explicitly in the formula for the mutual information.

Mutual Information

In general, in information theory the mutual information function measures the average amount of information that can be received by the user by observing the signal at the output of a channel [Bla90]. For the recall process of memorized patterns that we are discussing here, at each time step the process can be regarded as a channel with input ξ_i^μ and output $\sigma_{i,t}$ such that this mutual information function can be defined as (forgetting about the pattern index μ and the time index t) [BDA00]

$$I(\sigma_i; \xi_i) = S(\sigma_i) - \langle S(\sigma_i | \xi_i) \rangle_{\xi_i}; \quad (3.157)$$

$$S(\sigma_i) \equiv - \sum_{\sigma_i} p(\sigma_i) \ln[p(\sigma_i)], \quad (3.158)$$

$$S(\sigma_i | \xi_i) \equiv - \sum_{\sigma_i} p(\sigma_i | \xi_i) \ln[p(\sigma_i | \xi_i)]. \quad (3.159)$$

Here $S(\sigma_i)$ and $S(\sigma_i | \xi_i)$ are the entropy and the conditional entropy of the output, respectively. These information entropies are peculiar to the probability distributions of the output. The term $\langle S(\sigma_i | \xi_i) \rangle_{\xi_i}$ is also called the equivocation term in the recall process. The quantity $p(\sigma_i)$ denotes the probability distribution for the neurons at time t , while $p(\sigma_i | \xi_i)$ indicates the conditional probability that the i th neuron is in a state σ_i at time t , given that the i th pixel of the memorized pattern that is being retrieved is ξ_i . Hereby we have assumed that the conditional probability of all the neurons factorizes, i.e.,

$$p(\{\sigma_i\} | \{\xi_i\}) = \prod_i p(\sigma_i | \xi_i),$$

which is a consequence of the mean-field theory character of the presented model.⁶

The calculation of the different terms in the expression (3.157) proceeds as follows. Formally writing

$$\langle O \rangle \equiv \langle O \rangle_{\sigma | \xi, \xi} = \sum_{\xi} p(\xi) \sum_{\sigma} p(\sigma | \xi) O$$

⁶ A similar factorization has also been used in [SSP96].

for an arbitrary quantity O the conditional probability can be obtained in a rather straightforward way by using the complete knowledge about the system:

$$\begin{aligned} \langle \xi \rangle &= 0, & \langle \sigma \rangle &= 0, & \langle \sigma \xi \rangle &= am, & \langle \xi^2 \rangle &= a, & \langle \sigma^2 \rangle &= q, \\ \langle \sigma^2 \xi \rangle &= 0, & \langle \sigma \xi^2 \rangle &= 0, & \langle \sigma^2 \xi^2 \rangle &= an, & \langle 1 \rangle &= 1. \end{aligned}$$

The result reads (we forget about the index i)

$$\begin{aligned} p(\sigma|\xi) &= (s_\xi + m\xi\sigma)\delta(\sigma^2 - 1) + (1 - s_\xi)\delta(\sigma), \\ s_\xi &\equiv s_0 - \frac{q-n}{1-a}\xi^2, & s_0 &\equiv \frac{q-an}{1-a}. \end{aligned} \quad (3.160)$$

Alternatively, one can simply verify that this probability satisfies the averages

$$m = \frac{1}{a} \langle \langle \sigma \xi \rangle_{\sigma|\xi} \rangle_\xi, \quad q = \langle \langle \sigma^2 \rangle_{\sigma|\xi} \rangle_\xi, \quad n = \frac{1}{a} \langle \langle \sigma^2 \xi^2 \rangle_{\sigma|\xi} \rangle_\xi. \quad (3.161)$$

These averages are precisely equal in the limit $N \rightarrow \infty$ to the parameters m and q in (3.152)–(3.153) and to the activity–overlap introduced in (3.156). Using the probability distribution of the memorized patterns (3.150), we furthermore get

$$p(\sigma) \equiv \sum_\xi p(\xi)p(\sigma|\xi) = q\delta(\sigma^2 - 1) + (1 - q)\delta(\sigma). \quad (3.162)$$

The expressions for the entropies defined above become

$$\begin{aligned} S(\sigma) &= -q \ln \frac{q}{2} - (1 - q) \ln(1 - q), \\ \langle S(\sigma|\xi) \rangle_\xi &= -(1 - a) \left[s_0 \ln \frac{s_0}{2} + (1 - s_0) \ln(1 - s_0) \right] \\ &\quad - a \left[\frac{n+m}{2} \ln \frac{n+m}{2} + \frac{n-m}{2} \ln \frac{n-m}{2} + (1 - n) \ln(1 - n) \right] \end{aligned}$$

Self–Control Dynamics

General Equations

It is standard knowledge (e.g., [DGZ87, BSV94]) that the synchronous dynamics for diluted architectures can be solved exactly following the method based upon a signal–to–noise analysis of the local field (3.147) (e.g., [Ama77, BSV94, Oka96] and references therein). Without loss of generality we focus on the recall of one pattern, say $\mu = 1$, meaning that only $m_{N,t}^1$ is macroscopic, i.e., of order 1 and the rest of the patterns cause a cross–talk noise at each time step of the dynamics.

Supposing that the initial state of the network model $\{\sigma_{i,0}\}$ is a collection of IIDRV with mean zero and variance q_0 and correlated only with memorized

pattern 1 with an overlap m_0^1 it is wellknown (see references cited above) that the local field (3.147) converges in the limit $C, N \rightarrow \infty$ to [BDA00]

$$h_{i,0} = \xi_i^1 m_0^1 + [\alpha q_0]^{1/2} \mathcal{N}(0,1), \quad (3.163)$$

where the convergence is in distribution and where the quantity $\mathcal{N}(0,1)$ is a Gaussian random variable with mean zero and variance unity. The parameters m and q defined in the preceding sections have to be considered over the diluted structure and the (finite) loading α is defined by $p = \alpha C$.

This allows us to derive the first time-step in the evolution of the network. For diluted architectures this first step dynamics describes the full time evolution and we arrive at [DGZ87, BSV94]

$$m_{t+1}^1 = \int_{-\infty}^{+\infty} Dz \frac{1}{a} \langle \xi^1 F_{\theta_t}(\xi^1 m_t^1 + [\alpha q_t]^{1/2} z) \rangle, \quad (3.164)$$

$$q_{t+1} = \int_{-\infty}^{+\infty} Dz \langle F_{\theta_t}^2(\xi^1 m_t^1 + [\alpha q_t]^{1/2} z) \rangle. \quad (3.165)$$

where we recall that the $\langle \dots \rangle$ denote the average over the distribution of the memorized patterns and

$$Dz = dz [\exp(-z^2/2)] / (2\pi)^{1/2}.$$

Furthermore we also have the following expression for the activity-overlap

$$n_{t+1} = \frac{1}{a} \int_{-\infty}^{+\infty} Dz \langle (\xi^1)^2 F_{\theta_t}^2(\xi^1 m_t^1 + [\alpha q_t]^{1/2} z) \rangle. \quad (3.166)$$

For the specific transfer function defined in (3.148) the evolution equations (3.164)–(3.165) reduce to [BDA00]

$$m_{t+1} = \operatorname{erfc}\left(\frac{\theta_t - m_t}{\sqrt{\alpha q_t}}\right) - \operatorname{erfc}\left(\frac{\theta_t + m_t}{\sqrt{\alpha q_t}}\right), \quad (3.167)$$

$$q_{t+1} = a[\operatorname{erfc}\left(\frac{\theta_t - m_t}{\sqrt{\alpha q_t}}\right) + \operatorname{erfc}\left(\frac{\theta_t + m_t}{\sqrt{\alpha q_t}}\right)] + 2(1-a)[\operatorname{erfc}\left(\frac{\theta_t}{\sqrt{\alpha q_t}}\right)], \quad (3.168)$$

where we have dropped the index 1 and with the function $\operatorname{erfc}(\cdot)$ defined as

$$\operatorname{erfc}(x) \equiv \frac{1}{\sqrt{2\pi}} \int_x^\infty dz e^{-z^2/2}. \quad (3.169)$$

Without self-control these equations have been studied (see, e.g., [BSV94]).

Furthermore the first term in (3.168) gives the activity-overlap. More explicitly

$$n_{t+1} = \operatorname{erfc}\left(\frac{\theta_t - m_t}{\sqrt{\alpha q_t}}\right) + \operatorname{erfc}\left(\frac{\theta_t + m_t}{\sqrt{\alpha q_t}}\right). \quad (3.170)$$

Of course, it is known that the quality of the recall process is influenced by the cross-talk noise at each time step of the dynamics. A novel idea is then to let the network itself autonomously counter this cross-talk noise at each time step by introducing an adaptive, hence time-dependent, threshold of the form

$$\theta_t = c(a)\sqrt{\alpha q_t}. \quad (3.171)$$

Together with (3.164)–(3.166), this relation describes the self-control dynamics of the network model. For the present model with ternary neurons, this dynamical threshold is a macroscopic parameter, thus no average must be taken over the microscopic random variables at each time step t .⁷

This self-control mechanism is complete if we find a way to determine $c(a)$. Intuitively, looking at the evolution equations (3.164)–(3.165) after the straightforward average over ξ^1 has been done and requiring that

$$m \sim 1 - \operatorname{erfc}(n_s) \quad \text{and} \quad q \sim a + \operatorname{erfc}(n_s) \quad \text{with} \quad n_s > 0$$

inversely proportional to the error, the value of $c(a)$ should be such that mostly the argument of the transfer function satisfies

$$m - n_s\sqrt{\alpha q} \geq \theta \quad \text{and} \quad n_s\sqrt{\alpha q} \leq \theta.$$

This leads to $c \sim n_s$.⁸

Sparsely Coded Models

In the limit of *sparse coding* (see [WBL69, Ama89, Oka96] and references therein) meaning that the pattern activity a is very small and tends to zero for $N \rightarrow \infty$ it is possible to determine more precisely the factor $c(a)$ in the threshold (3.171).

We start from the evolution equations (3.167)–(3.168) governing the dynamics. To have $m \sim 1$ and $n \sim 1$ such that good recall properties, i.e., $\sigma_i = \xi_i$ for most i are realized, we want $m - \sqrt{\alpha q} \gg \theta$. Activity control, i.e., $q \sim a$ requires $\sqrt{\alpha q} \ll \theta$. From (3.171) we then get

$$1 \ll c(a) \ll \frac{1}{\sqrt{\alpha a}} - 1.$$

Then, for $c(a) \gg 1$ the second term on the right-hand side of (3.168) leads to [BDA00]

⁷ This is different from the idea used in some existing binary neuron models (see, e.g., [HD89]), where a local threshold $\theta_{i,t}$ is taken to study periodic behavior of the memorized patterns etc. Here we have in fact a mapping with a threshold changing each time step, but no statistical history intervenes in this process.

⁸ n_s itself depends on the loading α in the sense that for increasing α it gets more difficult to have good recall such that n_s decreases. But it can still be chosen *a priori*.

$$(1 - a)\text{erfc}[c(a)] \rightarrow \frac{1}{c(a)\sqrt{2\pi}} \exp\left[-\frac{c(a)^2}{2}\right]. \quad (3.172)$$

This term must vanish faster than a so that we get $c(a) = \sqrt{-2\ln(a)}$. This turns out to be the same factor as in the model with binary neurons [DB98, KA98]. Such a time-dependent threshold has also been used in binary models but for the recall of sparsely coded sequential patterns in the framework of statistical neurodynamics [AM88, Oka95] with the assumption that the temporal correlations up to the initial time can be neglected in the limit of an infinite number of neurons.

At this point we remark that in the limit of sparse coding, (3.167)–(3.170) for the overlap and the activity-overlap become

$$m_t \sim n_t \sim 1 - \frac{1}{2}\text{erfc}\left(\frac{m}{\sqrt{\alpha a}} - c\right). \quad (3.173)$$

Using all this and technically replacing the conditions $\ll$ above by $<$, meaning that we relax the requirement of perfect recall, we can evaluate the critical capacity for which some small errors in the recall process are allowed. We find

$$\alpha_c = O(|a \ln(a)|^{-1}), \quad (3.174)$$

which is of the same order as the critical capacity for binary sparsely coded network models with and without self-control (see [BDA00] and references therein).

Next we turn to the quality of the recall process by the network. Because of the sparse coding the Hamming distance is not a good measure since it does not distinguish between a situation where most of the wrong neurons are inactive and a situation where these wrong neurons are active. The errors in recalling the active states are much more relevant since they contain more information. For instance, when $\sigma_i = 0$ for all neurons the Hamming distance $d = a$ and vanishes in the limit of sparse coding, while when $\sigma_i = 1$ for all neurons it is $d = 1 - a$ and goes to 1. Clearly, in both cases no information is transmitted. Furthermore, suppose that all neurons are inactive, i.e., $\sigma_i = 0$, then we have that $m = 0$, $q = 0$ (and $n = 0$), so the Hamming distance is $d = a$ but there is no information transmitted. If instead of turning off the aN active neurons (meaning that $\xi_i = \pm 1$) one would turn on aN neurons among the inactive ones (meaning that $\xi_i = 0$) one would get $m = 1$, $q = 2a$ (and $n = 1$). So the Hamming distance is still $d = a$, but now some information is transmitted. It is intuitively clear that the first kind of action erases all the meaningful bits, while the second one does not affect essentially the code and, hence, leads to less important errors.

In fact, we immediately note that for the first example I is, indeed, 0. In the second example we find that $I = -a \ln(2a) - (1 - a) \ln(1 - a) = S(\xi) - 2a \ln(2)$ which is not much smaller than the entropy $S(\xi)$ of the memorized patterns. This confirms our statement that the mutual information is to be preferred

above the Hamming distance for discussing the quality of recall by sparsely coded network models [BDA00].

3.2.9 BAM and Statistical Neurodynamics

Recall that *bidirectional associative memory* (BAM) [Kos88] is a kind of an AM-model which is an artificial neural network. The principle function of AM is to memorize multiple patterns and to retrieve the correct one when a pattern key is given.

Also, recall that *autocorrelation associative memory* (AAM), sometimes called the *Hopfield model* [HT85], is also a kind of AM. AAM tries to retrieve a stored pattern when a degraded pattern is given as an association key; this type of retrieving is called homogeneous association. In contrast, BAM stores multiple pattern pairs and tries to retrieve a complete stored pattern pair when a degraded piece of the pair is given as an association key. Recall that in the association process, each component of a stored pattern pair drives BAM to retrieve themselves in a coordinate manner. Thus, BAM is called a *heterogeneous pattern association model* [SKO04].

In the field of neural networks, many efforts have been made to improve BAM from the viewpoint of computer application [HY89, Sim90, WCM91, ZHC93, OK94, WD95, Wan96, SZZ98], and few theoretical studies have been reported [HH-N88] [YSY91, TKK00]. The theoretical analysis of BAM has evolved with a focus on storage capacity, which means how many patterns can be stored in a network consisted of N neural units. Yanai et al. suggested that BAM can be regarded as a variation of AAM in which connections are systematically reduced [YSY91]. They also showed that the relative storage capacity, in which a finite amount of retrieval error is allowed, of BAM to be around $0.22N$. Haines and Hecht-Nielsen analyzed BAM along the same way, and reported its absolute capacity, in which no retrieval error is allowed, to be $O(N/\log N)$ [HH-N88]. Tanaka et al. analyzed BAM using a replica method, which is a statistical-mechanical analysis method, and showed its relative capacity to be $0.1998N$ [TKK00]. These studies mainly focused on the equilibrium state of BAM, and the transient process of retrieving, which means how to reach the equilibrium state, was not so conducted. However, analysis of the retrieval process is as important as that of the equilibrium state.

In this section following [SKO04], we analyze the equilibrium state of BAM using the self-consistent signal-to-noise analysis (SCSNA) [SF92], which is known as the cavity method. We found that the relative capacity was $0.1998N$, which agrees with the result of Tanaka et al. We also consider the retrieval process of BAM; we derived macroscopic dynamical equations using the statistical neurodynamics, which was theoretically derived in the same manner as the SCSNA [AM88, Oka95], and compared the results of between the statistical neurodynamics with those of computer simulation. Applying the statistical neurodynamics to BAM, we obtained the evolution equations for

the macroscopic parameters. In the limit of these evolution equations, that is, macroscopic parameters of BAM reached the equilibrium state, we found these values were consistent with the results of SCSNA. We also compared the results of applying the statistical neurodynamics with those of the computer simulation and obtained quantitative support for our analysis.

BAM Dynamics

As shown in Figure 3.3, BAM is a two-layered AM neural-network model [Kos88]. The first layer consists of cN neural units ($c \sim \mathcal{O}(1)$), and the state of the layer is denoted as x with the components denoted as x_i ($1 \leq i \leq cN$). The state of the second layer, which has $\tilde{c}N$ units, is denoted as $\tilde{x}$, and the j th unit state is described as $\tilde{x}_j$ ($1 \leq j \leq \tilde{c}N$). Each layer is connected by inter-layer connection J with the components described as J_{ij} ($1 \leq i \leq cN, 1 \leq j \leq \tilde{c}N$). J_{ij} represents the connection weight between the first layer unit x_i and the second layer unit $\tilde{x}_j$.

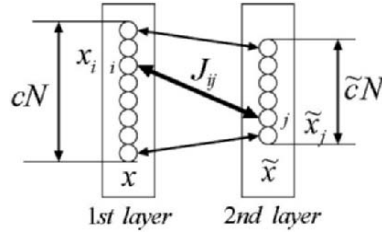


Fig. 3.3. Network structure of BAM (adapted and modified from [SKO04]).

Following [SKO04], we prepare p binary pattern pairs denoted by $\{\xi^\mu, \tilde{\xi}^\mu\}$ ($\mu = 1, \dots, p$), where the superscript μ denotes the pattern pair index. Pattern vector ξ^μ corresponds to the first layer, and $\tilde{\xi}^\mu$ corresponds to the second layer. Thus ξ^μ and $\tilde{\xi}^\mu$ have cN and $\tilde{c}N$ components, respectively, and each component, which is described as ξ_i^μ and $\tilde{\xi}_j^\mu$ ($1 \leq i \leq cN, 1 \leq j \leq \tilde{c}N$), respectively, is generated from uniform i.i.d.,

$$\Pr[\xi_i^\mu = \pm 1] = \frac{1}{2}, \quad (3.175)$$

$$\Pr[\tilde{\xi}_j^\mu = \pm 1] = \frac{1}{2}. \quad (3.176)$$

Assuming the number of stored pattern pairs to be $p \sim \mathcal{O}(N)$, we define the *loading rate* quantity $\alpha = p/N$, which controls the amount of pattern pairs to be stored in, so α is a *capacity index*.

To determine the inter-layer connection weight J_{ij} , which connects x_i and $\tilde{x}_j$, we use a *correlation-based learning* rule:

$$J_{ij} = \frac{1}{N} \sum_{\mu=1}^{\alpha N} \xi_i^\mu \tilde{\xi}_j^\mu. \quad (3.177)$$

$\xi^\mu, \tilde{\xi}^\mu$ ($\mu = 1, \dots, \alpha N$) are pattern pairs for association and the superscripts μ denotes the pattern pair index. All the pattern pair correlations between ξ^μ and $\tilde{\xi}^\mu$ are embedded in connection weight J . In this notation, the connections are not symmetrical, that is $J_{ij} \neq J_{ji}$.

In the retrieving, we use a synchronous update rule for each layer; that is, all units in each layer are updated synchronously, and these layers are updated alternately. The rules for updating the i th unit in the first layer and the j th unit in the second layer are

$$x_i^{2t} = F\left(\sum_{j=1}^{\tilde{c}N} J_{ij} \tilde{x}_j^{2t-1}\right), \quad (3.178)$$

$$\tilde{x}_j^{2t+1} = F\left(\sum_{i=1}^{cN} J_{ij} x_i^{2t}\right), \quad (3.179)$$

where t is the one-step time and $F(\cdot)$ is the smooth output sigmoid function, such as $\tanh(\cdot)$.

In these formulations, the retrieval process is carried out as follows. In an initial state, $t = 0$, an association key $x^0 = \{x_i^0\}$ is applied to the first layer. Then, all of the second layer units, $\tilde{x}_j^1$ ($1 \leq j \leq \tilde{c}N$), are updated using (3.179), and the state of the second layer is described as $\tilde{x}^1$. Next, at $t = 1$, all the units in the first layer, x_i^2 ($1 \leq i \leq cN$), are updated using (3.178), and the state is described as x^2 . After that the second layer is updated by (3.179). For each $t = 2, 3, \dots$, the alternate updating of each layer is carried out in the same way, and each layer state is denoted as x^{2t} and $\tilde{x}^{2t+1}$, for all $1 \leq i \leq cN$. In the next step, $2t - 1 = 1$, the whole units in the second layer $\tilde{x}_j^1$ are updated by (3.179). This alternate update is a characteristic of BAM.

To apply S/N analysis, we introduce overlaps, which means similarities between patterns. The overlaps between first layer state x^{2t} and the μ th pattern, ξ^μ , and between second-layer state $\tilde{x}^{2t-1}$ and $\tilde{\xi}^\mu$ are described as follows, respectively

$$m_\mu^{2t} = \frac{1}{cN} \sum_{i=1}^{cN} x_i^{2t} \xi_i^\mu, \quad \tilde{m}_\mu^{2t+1} = \frac{1}{\tilde{c}N} \sum_{j=1}^{\tilde{c}N} \tilde{x}_j^{2t+1} \tilde{\xi}_j^\mu.$$

Following the S/N analysis, we decomposed the inner term of $F(\cdot)$ in (3.178) and (3.179) into signal and noise components. Assuming the first pattern pair, $\{\xi^1, \tilde{\xi}^1\}$, is retrieved, the terms including overlaps m_1 and $\tilde{m}_1$, which mean how well the first pattern pair is retrieved are signal components, i.e., $m_1, \tilde{m}_1 \sim \mathcal{O}(1)$. Using these overlaps, (3.178) and (3.179) can be described as

$$x_i^{2t} = F(\tilde{c}\tilde{m}_1^{2t-1}\xi_i^1 + z_i^{2t-1}), \quad (3.180)$$

$$\tilde{x}_j^{2t+1} = F(cm_1^{2t}\tilde{\xi}_j^1 + \tilde{z}_j^{2t}), \quad (3.181)$$

where z_i^{2t-1} , and $\tilde{z}_j^{2t}$ are called the *crosstalk noises*, which are the effects from other pattern pairs, $\{\xi^\mu, \tilde{\xi}^\mu\}$ ($\mu = 2, \dots, \alpha N$), which prevents the target pair $\{\xi^1, \tilde{\xi}^1\}$ to be retrieved. These crosstalk noises are defined by

$$z_i^{2t-1} = \frac{1}{N} \sum_{\mu=2}^{\alpha N} \sum_{j=1}^{\tilde{c}N} \xi_i^\mu \tilde{\xi}_j^\mu \tilde{x}_j^{2t-1}, \quad (3.182)$$

$$\tilde{z}_j^{2t} = \frac{1}{N} \sum_{\mu=2}^{\alpha N} \sum_{i=1}^{cN} \tilde{\xi}_j^\mu \xi_i^\mu x_i^{2t}. \quad (3.183)$$

Equilibrium State Analysis

To derive equilibrium state macroscopic parameters, we use SCSNA [SF92], which is an extension of a naive signal-to-noise (S/N) analysis. Since the SCSNA treats the equilibrium states of an AM-model, we omit the index t in the update rules. The time index corresponding to t is negligible in the updating rules (3.180) and (3.181). Therefore, each layer units satisfies

$$x_i = F\left(\sum_{j=1}^{\tilde{c}N} J_{ij} \tilde{x}_j\right) \quad \text{and} \quad \tilde{x}_j = F\left(\sum_{i=1}^{cN} J_{ij} x_i\right), \quad (3.184)$$

and we denote the vectors pair $\{x = \{x_i\}, \tilde{x} = \{\tilde{x}_j\}\}$ as each layer's state in the equilibrium state. Hence we can rewrite (3.180) and (3.181) as

$$x_i = F(\tilde{c}\tilde{m}_1\xi_i^1 + z_i), \quad \text{and} \quad (3.185)$$

$$\tilde{x}_j = F(cm_1\tilde{\xi}_j^1 + \tilde{z}_j), \quad (3.186)$$

respectively. In a naive S/N analysis, each noise term, z_i and $\tilde{z}_j$, is assumed to obey an independent identical Gaussian distribution w.r.t the site i and j , respectively, regarded as an i.i.d. random number. In contrast, the SCSNA treats z_i and $\tilde{z}_j$ more precisely [SF92].

In the SCSNA, the crosstalk noise term is decomposed into a systematic bias term and a Gaussian noise term with 0 mean [SF92]. The SCSNA evaluates infinitesimal effects came from a μ th pattern pair $\{\xi^\mu, \tilde{\xi}^\mu\}$, and the effects are appeared as the self-dependent components of x_i and $\tilde{x}_j$. The SCSNA [SF92] evaluates the effective self-depend components came from the ν th pattern pair, in the crosstalk noises. Taking these effects into consideration, we can derive the following self-consistent *order parameter equations*:

$$\begin{aligned}
Y &= F(\tilde{c}\tilde{m}\xi + \frac{\alpha\tilde{c}\tilde{U}}{1 - c\tilde{c}U\tilde{U}}Y + \sqrt{\alpha r}z), \\
\tilde{Y} &= F(cm\tilde{\xi} + \frac{\alpha cU}{1 - c\tilde{c}U\tilde{U}}\tilde{Y} + \sqrt{\alpha\tilde{r}}\tilde{z}), \\
m &= \int Dz : \langle \xi Y \rangle_{\xi}, \quad \tilde{m} = \int Dz \langle \tilde{\xi} \tilde{Y} \rangle_{\tilde{\xi}}, \\
q &= \int Dz \langle Y^2 \rangle_{\xi}, \quad \tilde{q} = \int Dz \langle \tilde{Y}^2 \rangle_{\tilde{\xi}}, \\
U &= \frac{1}{\sqrt{\alpha r}} \int Dz z \langle Y \rangle_{\xi}, \quad \tilde{U} = \frac{1}{\sqrt{\alpha\tilde{r}}} \int Dz z \langle \tilde{Y} \rangle_{\tilde{\xi}}, \\
r &= \frac{\tilde{c}}{(1 - c\tilde{c}U\tilde{U})^2} (\tilde{q} + c\tilde{c}\tilde{U}^2 q), \\
\tilde{r} &= \frac{c}{(1 - c\tilde{c}U\tilde{U})^2} (q + c\tilde{c}U^2 \tilde{q}).
\end{aligned} \tag{3.188}$$

These equations are described in the style of [SF92]. Y and $\tilde{Y}$ represents equilibrium states x_i and $\tilde{x}_j$, respectively. the effective outputs for x_i and $\tilde{x}_j$, respectively. The stochastic variables ξ and $\tilde{\xi}$, obeying (3.175) and (3.176), corresponds to a retrieving pattern components ξ_i^1 and $\tilde{\xi}_j^1$, and order parameters m and $\tilde{m}$ corresponds to overlaps m_1 and $\tilde{m}_1$. Note that the operators $\langle \cdot \rangle_{\xi}$ and $\langle \cdot \rangle_{\tilde{\xi}}$ mean the expectations for stochastic variables ξ or $\tilde{\xi}$, respectively.

These expectations come from the substitution of the averaging operation described as

$$\frac{1}{cN} \sum_{i=1}^{cN} \rightarrow \langle \cdot \rangle_{\xi}, \quad \text{and} \quad \frac{1}{\tilde{c}N} \sum_{j=1}^{\tilde{c}N} \rightarrow \langle \cdot \rangle_{\tilde{\xi}}.$$

The stored patterns ξ^{μ} and $\tilde{\xi}^{\mu}$ can be considered as stochastic variables which come from *independent and identical distribution* (i.i.d.).

In (3.187), each argument of the function $F(\cdot)$ consists of three parts. The first terms, $\tilde{c}\tilde{m}\xi$ and $cm\tilde{\xi}$, comes from the signal components, the second terms, $\frac{\alpha\tilde{c}\tilde{U}}{1 - c\tilde{c}U\tilde{U}}Y$ and $\frac{\alpha cU}{1 - c\tilde{c}U\tilde{U}}\tilde{Y}$, mean the systematic bias of the crosstalk noises (z_i , $\tilde{z}_j$) in (3.182) and (3.183), and each third term comes from the other crosstalk noise components. Each third term is assigned to be a Gaussian distribution with 0 mean and αr or $\alpha\tilde{r}$ variance.

Assuming each crosstalk noise except the self-dependent components, which we denote z'_i and $\tilde{z}'_j$, respectively, follows an identical independent normal distribution, we can evaluate the effect of these noises with a Gaussian integral:

$$\int Dz = \frac{1}{\sqrt{2\pi}} \int dz \exp\left(-\frac{z^2}{2}\right).$$

Normal distribution is described with a mean and a variance, and each noise (z'_i , $\tilde{z}'_j$) follows $N(0, \alpha r)$ and $N(0, \alpha\tilde{r})$, respectively. Thus the third terms

can be substituted by $\sqrt{\alpha r}z$ and $\sqrt{\alpha \tilde{r}}z$, respectively, where the stochastic variable z follows $N(0, 1)$. Both noise distribution means are equal to 0, and the variances are equal to αr and $\alpha \tilde{r}$, respectively.

BAM Retrieval Process

As we have seen, the SCSNA described the equilibrium state of BAM quantitatively. Here we consider a retrieval process of BAM, which means the transient process reaching the equilibrium state.

The *statistical neurodynamics*, an analysis method for transient process of neural networks, is actually a retrieval theory for AM models, based on S/N analysis of Amari and Maginu [AM88]. This theory assumes that each crosstalk noise obeys an identical independent Gaussian distribution, and this is called one-step analysis. Using the SCSNA concept, Okada improved the one-step analysis in order to evaluate crosstalk noise correlation precisely [Oka95]. The Okada's analysis method succeeded to explain the transient process of several neural networks quantitatively [KO02]. It is known that the storage capacity obtained by Amari and Maginu theory does not coincide with the results of the replica theory [AGS85b], and the size of the basin of attraction derived from Amari and Maginu theory is larger than the results of the computer simulation. Okada extended the Amari and Maginu theory to improve to resolve these difficulties [Oka95], and obtained a macroscopic equation which has hierarchical structure. In the macroscopic equation, the first-order approximation corresponds to the Amari and Maginu theory, and the higher order approximation coincide with the replica theory.

Here we derive the evolution equations for the macroscopic parameters of BAM in the manner of [Oka95], using the SCSNA extension of statistical neurodynamics.

For applying the statistical neurodynamics to BAM, we evaluate the crosstalk noises (3.182) and (3.183) in equations (3.180) and (3.181). Assuming the first pattern pair, $\{\xi^1, \tilde{\xi}^1\}$, is retrieved, we can regard the overlaps of other pattern pairs, $\{m_\mu^{2t}, \tilde{m}_\mu^{2t+1}\}$ where $\mu \geq 2$, as small. Thus, we expand the state x_i^{2t} and $\tilde{x}_j^{2t+1}$:

$$x_i^{2t} = x_i^{2t,(\mu)} + \tilde{c} \tilde{m}_\mu^{2t-1} \xi_i^\mu F'(\tilde{c} \sum_{\nu \neq \mu}^{\alpha N} \tilde{m}_\mu^{2t-1} \xi_i^\nu), \quad (3.189)$$

$$\begin{aligned} \tilde{x}_j^{2t+1} &= F(c \sum_{\nu \neq \mu}^{\alpha N} m_\nu^{2t} \tilde{\xi}_j^\nu) + c m_\mu^{2t} \tilde{\xi}_j^\mu F'(c \sum_{\nu \neq \mu}^{\alpha N} m_\nu^{2t} \tilde{\xi}_j^\nu) \\ &= \tilde{x}_j^{2t+1,(\mu)} + c m_\mu^{2t} \tilde{\xi}_j^\mu F'(c \sum_{\nu \neq \mu}^{\alpha N} m_\nu^{2t} \tilde{\xi}_j^\nu), \end{aligned} \quad (3.190)$$

for $\mu \geq 2$, where $x_i^{2t,(\mu)}$ and $\tilde{x}_j^{2t+1,(\mu)}$ mean the value drawn the effect of μ th pattern pair from $\tilde{x}_j^{2t+1}$ and x_i^{2t} , respectively, i.e.,

$$\tilde{x}_j^{2t+1,(\mu)} = F(c \sum_{\nu \neq \mu}^{\alpha N} m_\nu^{2t} \tilde{\xi}_j^\nu), \quad x_i^{2t,(\mu)} = F(\tilde{c} \sum_{\nu \neq \mu}^{\alpha N} \tilde{m}_\mu^{2t-1} \xi_i^\nu).$$

Substituting (3.189) and (3.190) into (3.182) and (3.183), we get,

$$z_i^{2t+1} = \alpha \tilde{c} \tilde{U}_{2t+1} x_i^{2t,(\mu)} + Z_i^{2t+1}, \quad \tilde{z}_j^{2t} = \alpha c U_{2t} \tilde{x}_j^{2t-1,(\mu)} + \tilde{Z}_j^{2t}, \quad \text{where}$$

$$Z_i^{2t+1} = \frac{1}{N} \sum_{\mu=2}^{\alpha N} \sum_{j=1}^{\tilde{c}N} \xi_i^\mu \tilde{\xi}_j^\mu \tilde{x}_j^{2t+1,(\mu)} + \frac{\tilde{c} \tilde{U}_{2t+1}}{N} \sum_{\mu=2}^{\alpha N} \sum_{k \neq i}^{cN} \xi_i^\mu \xi_k^\mu x_k^{2t,(\mu)} + \tilde{c} \tilde{U}_{2t+1} U_{2t} Z_i^{2t-1}, \quad (3.191)$$

$$\tilde{Z}_j^{2t} = \frac{1}{N} \sum_{\mu=2}^{\alpha N} \sum_{i=1}^{cN} \tilde{\xi}_j^\mu \xi_i^\mu x_i^{2t,(\mu)} + \frac{c U_{2t}}{N} \sum_{\mu=2}^{\alpha N} \sum_{l \neq j}^{\tilde{c}N} \tilde{\xi}_j^\mu \tilde{\xi}_l^\mu \tilde{x}_l^{2t-1,(\mu)} + c \tilde{c} U_{2t} \tilde{U}_{2t-1} \tilde{Z}_j^{2t-2}, \quad (3.192)$$

with

$$\tilde{U}_{2t+1} = \frac{1}{\tilde{c}N} \sum_{j=1}^{\tilde{c}N} F'(c \sum_{\nu \neq \mu}^{\alpha N} m_\nu^{2t} \tilde{\xi}_j^\nu), \quad U_{2t} = \frac{1}{cN} \sum_{i=1}^{cN} F'(\tilde{c} \sum_{\nu \neq \mu}^{\alpha N} \tilde{m}_\mu^{2t-1} \xi_i^\nu).$$

Since $x_i^{2t,(\mu)}$ and $\tilde{x}_j^{2t-1,(\mu)}$ are almost independent with ξ_i^μ and $\tilde{\xi}_j^\mu$, respectively, each Z_i^{2t+1} and $\tilde{Z}_j^{2t}$ can be regarded as independent identical Gaussian distributions, that is $Z_i^{2t+1} \sim N(0, \alpha r_{2t+1})$ and $\tilde{Z}_j^{2t} \sim N(0, \alpha \tilde{r}_{2t})$. Each noise variance,

$$E[(Z_i^{2t+1})^2] = \alpha r_{2t+1} \quad \text{and} \quad E[(\tilde{Z}_j^{2t})^2] = \alpha \tilde{r}_{2t},$$

can be defined by

$$\alpha r_{2t+1} = \alpha \tilde{c} \tilde{q}_{2t+1} + \alpha \tilde{c} \tilde{c}^2 \tilde{U}_{2t+1}^2 q_{2t} + \alpha (\tilde{c} \tilde{U}_{2t+1} U_{2t})^2 r_{2t-1} + 2c \tilde{c} \tilde{U}_{2t+1} U_{2t} E \left[Z_i^{2t-1} \frac{1}{N} \sum_{\mu=2}^{\alpha N} \sum_{j=1}^{\tilde{c}N} \xi_i^\mu \tilde{\xi}_j^\mu \tilde{x}_j^{2t+1,(\mu)} \right], \quad (3.193)$$

$$\alpha \tilde{r}_{2t} = \alpha c q_{2t} + \alpha \tilde{c} c^2 U_{2t}^2 \tilde{q}_{2t-1} + \alpha (c \tilde{c} U_{2t} \tilde{U}_{2t-1})^2 \tilde{r}_{2t-2} + 2c \tilde{c} U_{2t} \tilde{U}_{2t-1} E \left[\tilde{Z}_j^{2t-2} \frac{1}{N} \sum_{\mu=2}^{\alpha N} \sum_{i=1}^{cN} \tilde{\xi}_j^\mu \xi_i^\mu x_i^{2t,(\mu)} \right], \quad \text{where} \quad (3.194)$$

$$\tilde{q}_{2t+1} = \frac{1}{\tilde{c}N} \sum_{j=1}^{\tilde{c}N} \left(\tilde{x}_j^{2t+1,(\mu)} \right)^2, \quad q_{2t} = \frac{1}{cN} \sum_{i=1}^{cN} \left(x_i^{2t,(\mu)} \right)^2.$$

The last terms in (3.193) and (3.194) are determined by correlations between the current state $\tilde{x}_j^{2t+1}$ and the previous state noise variable Z_i^{2t-1} , and between x_i^{2t} and $\tilde{Z}_j^{2t-2}$, respectively. Assuming that the $(n+1)$ previous state

noise variables $Z_i^{2(t-n)-1}$ and $\tilde{Z}_j^{2(t-n)-2}$ have no correlation with the current state $\tilde{x}_j^{2t+1}$ and x_i^{2t} , respectively, we can expand r_{2t+1} and $\tilde{r}_{2t}$ in the following recurrence formulas:

$$\begin{aligned} r_{2t+1} = & \tilde{c}\tilde{q}_{2t+1} + c(\tilde{c}\tilde{U}_{2t+1})^2 q_{2t} + (c\tilde{c}\tilde{U}_{2t+1}U_{2t})^2 r_{2t-1} \\ & + 2\tilde{c} \sum_{\eta=1}^n (c\tilde{c})^\eta \tilde{q}_{2t+1,2(t-\eta)+1} \prod_{\tau=t-\eta+1}^t \tilde{U}_{2\tau+1} U_{2\tau} \\ & + 2c(\tilde{c}\tilde{U}_{2t+1})^2 \sum_{\eta=1}^{n-1} (c\tilde{c})^\eta q_{2t,2(t-\eta)} \prod_{\tau=t-\eta+1}^t U_{2\tau} \tilde{U}_{2\tau-1}, \end{aligned} \quad (3.195)$$

$$\begin{aligned} \tilde{r}_{2t} = & cq_{2t} + \tilde{c}(cU_{2t}^2)^2 \tilde{q}_{2t-1} + (\tilde{c}cU_{2t}\tilde{U}_{2t-1})^2 \tilde{r}_{2t-2}, \\ & + 2c \sum_{\eta=1}^n (\tilde{c}c)^\eta q_{2t,2(t-\eta)} \prod_{\tau=t-\eta+1}^t U_{2\tau} \tilde{U}_{2\tau-1} \\ & + 2\tilde{c}(cU_{2t})^2 \sum_{\eta=1}^{n-1} (\tilde{c}c)^\eta \tilde{q}_{2t-1,2(t-\eta)-1} \prod_{\tau=t-\eta+1}^t \tilde{U}_{2\tau-1} U_{2\tau-2}, \end{aligned} \quad (3.196)$$

where $\tilde{q}_{2t+1,2(t-n)+1}$ means a cross-correlation between the current state $\tilde{x}_j^{2t+1}$ and the n -step previous state $\tilde{x}_j^{2(t-n)+1}$, and $q_{2t,2(t-n)}$ means a cross-correlation between x_i^{2t} and $x_i^{2(t-n)}$. These variables can be also described with the macroscopic parameters across the n -step previous state. The complete formula is described in the appendix.

Assuming the self-averaging property, we get the evolution equations for macroscopic parameters as follows:

$$\begin{aligned} Y^{2t} &= F(\tilde{c}\tilde{m}_{2t-1}\xi + \sqrt{\alpha r_{2t-1}}z), \\ \tilde{Y}^{2t+1} &= F(cm_{2t}\tilde{\xi} + \sqrt{\alpha \tilde{r}_{2t}}z), \\ m^{2t} &= \int Dz \langle \xi Y^{2t} \rangle_\xi, \quad \tilde{m}^{2t+1} = \int Dz \langle \tilde{\xi} \tilde{Y}^{2t+1} \rangle_{\tilde{\xi}}, \\ q_{2t} &= \int Dz \langle (Y^{2t})^2 \rangle_\xi, \quad \tilde{q}_{2t+1} = \int Dz \langle (\tilde{Y}^{2t+1})^2 \rangle_{\tilde{\xi}}, \\ U_{2t} &= \frac{1}{\sqrt{\alpha r_{2t-1}}} \int Dz z \langle Y^{2t} \rangle_\xi, \quad \tilde{U}_{2t+1} = \frac{1}{\sqrt{\alpha \tilde{r}_{2t}}} \int Dz z \langle \tilde{Y}^{2t+1} \rangle_{\tilde{\xi}}. \end{aligned} \quad (3.198)$$

In these order-parameter equations, Y^{2t} and $\tilde{Y}^{2t+1}$ correspond to the x_i^{2t} and $\tilde{x}_j^{2t+1}$, respectively. The overlaps for the first pattern pair, m_1^{2t} and $\tilde{m}_1^{2t+1}$, which mean retrieval degree, correspond to the m^{2t} and $\tilde{m}^{2t+1}$, respectively.

Analysis

Here, following [SKO04], we first apply the one-step analysis method [AM88], which corresponds to the naive S/N analysis. Then, we extend the one-step analysis to the statistical neurodynamics.

One-Step Theory. Recall that Amari and Maginu proposed an analysis method called ‘one-step theory’ [AM88], which is based on the naive S/N analysis, and they applied it to AAM. In the one-step theory, the noise components, which corresponds to (3.182) and (3.183) in each step, are assumed to be independent Gaussian noise.

The important point here is that the evolution of macroscopic parameters can be described as the recurrence formulae using one-step previous state. These recurrence formulae can be described with the macroscopic parameter of the one-step before state. Formally, these equations are identical to the analysis of the sequence association model, which is a variety of AAM [Ama89]. In the sequence association model, since cross correlations between sequential patterns are embedded in the connections, the stored patterns appear sequentially in each update in the successful retrieving phase.

Critical capacity of the limit of the above dynamics was derived in [SKO04] and obtained $\alpha_c = 0.27$, which had been suggested by Amari [Ama88]. However, the critical capacity derived from the one-step theory seems to be overestimated. This overestimation comes from the assumption that the noise components in each time step are independent Gaussian noise. In other words, we should precisely evaluate the parameters r_{2t+1} , and $\tilde{r}_{2t}$ have no correlation to the previous state in each update.

To evaluate the noise correlation in time series more accurately, we must introduce the statistical neurodynamics. Therefore, we need to evaluate these noise correlations exactly in accordance with the concept of SCSNA.

Statistical Neurodynamics. Just like one-step analysis is based on the naive S/N analysis, the statistical neurodynamics analysis is based on the SCSNA. This analysis evaluates the noise correlation more accurately than does the one-step analysis. Using the statistical neurodynamics, Okada described the transient process of an AAM as recurrence formulae of macroscopic parameters [Oka95].

To apply the statistical neurodynamics to BAM, only two parameters, r_{2t+1} and $\tilde{r}_{2t}$, need to be evaluated, and the other parameters are the same as those of the one-step analysis. There was no need to re-evaluate the other parameters $Y_i^{2t}, \tilde{Y}_j^{2t+1}, m_{2t}^1, \tilde{m}_{2t+1}^1, q_{2t}, \tilde{q}_{2t+1}, U_{2t}, \tilde{U}_{2t+1}$.

Yanai et al. [YSY91] applied the one-step analysis, which corresponds to Amari and Maginu theory. In their analysis, the macroscopic order parameter equations for $Y_i^{2t}, \tilde{Y}_j^{2t+1}, m_1^{2t}, \tilde{m}_1^{2t+1}, q_{2t}, \tilde{q}_{2t+1}, U_{2t}, \tilde{U}_{2t+1}$, which are described in (3.197)–(3.198), are identical to those of our analysis. The differences are in evaluating of noise variances, that is, r_{2t+1} and $\tilde{r}_{2t}$. They ignored the noise correlation, and derived these values as

$$r_{2t+1} = \tilde{c}\tilde{q}_{2t+1} + c(\tilde{c}\tilde{U}_{2t+1})^2 q_{2t}, \quad (3.199)$$

$$\tilde{r}_{2t} = c q_{2t} + \tilde{c}(c U_{2t}^2)^2 \tilde{q}_{2t-1}. \quad (3.200)$$

In their result, the critical capacity α_c is equal to 0.27, which is not equal to our SCSNA analysis and the replica analysis ($\alpha_c = 0.1998$ for both studies). This overestimation comes from the lack of noise correlation evaluation.

In our analysis, we consider the effect of crosstalk noise correlation across n -step previous state, and obtained the equations (3.193) and (3.194) which includes Yanais' analysis (3.199) and (3.200). Hereafter, we call the statistical neurodynamics considering across the n -step previous state as the ' n -step analysis'. 'Full-step' analysis means using all the macroscopic parameters from the initial state ($t = 0$) to the current state.

In (3.191) and (3.192), the first two terms are also in the one-step analysis. When we truncate the expansion with $n = 1$, these parameters r_{2t+1} and $\tilde{r}_{2t}$ are identical to those in the one-step analysis, meaning that the statistical neurodynamics includes the one-step analysis, and the residual terms describes higher order correlations. Since these cross-correlations can be denoted as the recurrence formulae using across n time steps before states, the computational cost for analysis is not so much high. The one-step analysis is identical to the statistical neurodynamics analysis of $n = 1$. For more details, see [SKO04].

3.2.10 Firing Correlation in Layered AM Neural Nets

Recall that Abeles proposed a *synfire chain* as a model of the cortex [Abe91]. It consists of neuron pools connected to each other in a feedforward chain so that firings can propagate. Repeated spatiotemporal firing patterns with precise timing accuracy within milliseconds have been recorded in the frontal cortex of monkeys performing tasks [ABM93]. These findings suggest the occurrence of some temporal structure in the brain. In recent experiments, temporally precise firing sequences of spikes have been reported for a songbird vocal control system [PVB98, HZF02]. It was also proved that sequences of synchronous activity propagate through a reproduced multilayer feedforward network of neurons in an in vitro slice preparation of rat cortex [KTD03, Rey03, IAC04].

Models of a *synfire chain* and its mechanism have been studied theoretically by many researchers (see [Bie95, HHP95, HP96, DGA99, CF01, ANW03, HOY04, HOY05, KYO04]). Recently, Diesmann and Cateau et al. investigated conditions for stable propagation of correlated firings along the feedforward networks [DGA99, CF01].

Amari et al. [ANW03] analyzed a model of a neuron pool in which neurons receive common overlapping input, and proved that higher order interactions of neurons exist in synchronous firing, which generates wide-spread activity distributions. That is, the self-averaging property breaks down even in the stochastic limit, and activity depends on the sample, therefore activity distribution is not concentrated on its mean value and has a wide-spread distribution. However, in their model, neurons are uniformly connected to each other and each connection has no structures such as learning efficacies.

In this subsection, following [YO05], we investigate a layered associative memory NN-model in which connections patterns are embedded together with uniform noise which induces common input into the next layer. Common input in layers generates correlated firing of neurons. We discuss cases of infinite pattern loading in the networks.

A Layered AM-Model

We consider a layered AM model. In this model, neuron states are denoted by x_i^ℓ (neuron at i site in layer ℓ), which takes $+1$ or -1 corresponding to the firing state and non-firing state, respectively. Firing of neurons is propagated according to [YO05]

$$x_i^{\ell+1} = F(J_{ij}^\ell x_j^\ell), \quad (3.201)$$

where the layer number is $\ell = 1, \dots, L$ and the number of neurons in a layer is $i = 1, \dots, N$. F is the output function $F(h) = \text{sgn}(h)$. For the coupling of neurons, we consider the following form:

$$J_{ij}^\ell = \frac{1}{N} \xi_i^{\ell+1, \mu} \xi_j^{\ell, \mu} + w_j^\ell, \quad (3.202)$$

where, the first term describes correlated learning in which the patterns are embedded. Each component of the patterns is assumed to be an independent random variable that takes a value of either $+1$ or -1 according to the probability

$$\Pr[\xi_i^{\ell, \mu} = \pm 1] = \frac{1}{2}, \quad (\mu = 1, \dots, p, \ell = 0, \dots, L),$$

where the number of patterns is $p = \alpha N$. The second term of (3.202) is uniform Gaussian noise $w_j^\ell \sim \mathcal{N}(0, \delta^2/N)$. The coupling w_j^ℓ propagates a firing of x_j^ℓ to the next layer uniformly, so it induces *synfiring* in the next layer.

We introduce the order parameter characterizing memory retrieval, namely, the overlap with the μ th pattern in the ℓ th layer, defined by

$$m^{\ell, \mu} = \frac{1}{N} \xi_i^{\ell, \mu} x_i^\ell.$$

Order-Parameter Equations

In the following analysis, we consider the situation where the pattern $\mu = 1$ is being retrieved, i.e., $m^{\ell, 1} \equiv m^\ell \sim \mathcal{O}(1)$, and $m^{\ell, \mu \neq 1} \sim \mathcal{O}(1/\sqrt{N})$. In this case, the input to the $\ell + 1$ th layer can be written as [YO05]

$$h_i^\ell \equiv J_{ij}^\ell x_j^\ell = \xi_i^{\ell+1} m^\ell + z_i^\ell + \eta^\ell, \quad \text{where} \quad (3.203)$$

$$z_i^\ell \equiv \frac{1}{N} \xi_i^{\ell+1, \mu} \xi_j^{\ell, \mu} x_j^\ell, \quad \eta^\ell \equiv w_j^\ell x_j^\ell. \quad (3.204)$$

The first term of (3.203) is a signal term, the second term is the cross-talk noise term, and the last term is the uniform noise term $\eta^\ell \sim \mathcal{N}(0, \delta^2)$, which is independent of i .

The evolution of the neuron states then is given by

$$x_i^{\ell+1} = F(h_i) = F(\xi_i^{\ell+1} m^\ell + z_i^\ell + \eta^\ell). \quad (3.205)$$

We note that in (3.205), η^ℓ acts as a uniform threshold, so it induces synfiring in the $\ell + 1$ th layer.

Let us now derive equations describing the evolution of the order parameter m^ℓ . We can then follow the standard procedure [Oka95] to expand (3.204) as

$$\begin{aligned} z_i^{\ell+1} &= \frac{1}{N} \xi_i^{\ell+2, \mu} \xi_j^{\ell+1, \mu} F[\xi_j^{\ell+1} m^\ell + \eta^\ell + \frac{1}{N} \xi_j^{\ell+1, \mu} \xi_k^{\ell, \mu} x_k^\ell + \frac{1}{N} \xi_j^{\ell+1, \nu} \xi_k^{\ell, \nu} x_k^\ell] \\ &\simeq \frac{1}{N} \xi_i^{\ell+2, \mu} \xi_j^{\ell+1, \mu} F[\xi_j^{\ell+1} m^\ell + \eta^\ell + \frac{1}{N} \xi_j^{\ell+1, \nu} \xi_k^{\ell, \nu} x_k^\ell] \\ &\quad + \frac{1}{N^2} \xi_i^{\ell+2, \mu} \xi_k^{\ell, \mu} x_k^\ell F'[\xi_j^{\ell+1} m^\ell + \eta^\ell + \frac{1}{N} \xi_j^{\ell+1, \nu} \xi_k^{\ell, \nu} x_k^\ell]. \end{aligned}$$

We then get

$$z_i^{\ell+1} = \frac{1}{N} \xi_i^{\ell+2, \mu} \xi_j^{\ell+1, \mu} x_j^{\ell+1, (\mu)} + \sum_{n=0}^L \left(\prod_{t=0}^n U^{\ell+1-t} \right) \frac{1}{N} \xi_i^{\ell+2, \mu} \xi_j^{\ell-n, \mu} x_j^{\ell-n, (\mu)},$$

where we have introduced the following two quantities [YO05]:

$$\begin{aligned} x_j^{\ell+1, (\mu)} &= F[\xi_j^{\ell+1} m^\ell + \eta^\ell + \frac{1}{N} \xi_j^{\ell+1, \nu} \xi_k^{\ell, \nu} x_k^\ell], \\ U^{\ell+1} &= \frac{1}{N} F'[\xi_j^{\ell+1} m^\ell + \eta^\ell + \frac{1}{N} \xi_j^{\ell+1, \nu} \xi_k^{\ell, \nu} x_k^\ell]. \end{aligned}$$

As usual, we assume that z_i^ℓ obeys a Gaussian distribution, where the mean equals 0 and the variance of z_i^ℓ , $E[(z_i^{\ell+1})^2] = (\sigma^\ell)^2$, is given by

$$\begin{aligned} (\sigma^{\ell+1})^2 &= \alpha [1 + (U^{\ell+1})^2 (U^\ell)^2 + (U^{\ell+1})^2 (U^\ell)^2 (U^{\ell-1})^2 + \dots] \\ &= \alpha \left[1 + (U^{\ell+1})^2 \frac{(\sigma^\ell)^2}{\alpha} \right] = \alpha + (U^{\ell+1})^2 (\sigma^\ell)^2. \end{aligned} \quad (3.206)$$

Finally, when m^ℓ, σ^ℓ and η^ℓ are given, we get the following coupled order parameter equations using $F(h) = \text{sgn}(h)$:

$$\begin{aligned}
m^{\ell+1}(m^\ell, \sigma^\ell, \eta^\ell) &= \int Dz^\ell \langle \xi^{\ell+1} F(\xi^{\ell+1} m^\ell + z^\ell + \eta^\ell) \rangle_\xi \\
&= \frac{1}{2} [\text{erf}(u) + \text{erf}(v)], \tag{3.207}
\end{aligned}$$

$$\begin{aligned}
U^{\ell+1}(m^\ell, \sigma^\ell, \eta^\ell) &= \int Dz^\ell \langle F'(\xi^{\ell+1} m^\ell + z^\ell + \eta^\ell) \rangle_\xi \\
&= \frac{1}{\sqrt{2\pi}\sigma^\ell} (e^{-u^2} + e^{-v^2}), \tag{3.208}
\end{aligned}$$

$$Dz^\ell = \frac{dz^\ell}{\sqrt{2\pi}\sigma^\ell} \exp\left(-\frac{(z^\ell)^2}{2(\sigma^\ell)^2}\right),$$

where $\langle \cdot \rangle_\xi$ denotes the average over random patterns of one layer,

$$u = (m^\ell + \eta^\ell)/(\sqrt{2}\sigma^\ell), \quad v = (m^\ell - \eta^\ell)/(\sqrt{2}\sigma^\ell), \quad \text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-y^2} dy.$$

Here, we use $\text{sgn}'(x) = 2\delta(x)$.

Probability Distribution Function

Let us now discuss the effect of η . In the presence of η^ℓ , the evolution of x_i^ℓ is given by (3.205). For a given η^ℓ , the order parameters at the $\ell + 1$ th layer are given in terms of those at the ℓ th layer as (3.206)–(3.208). However, η^ℓ is distributed in each layer ℓ according to the Gaussian distribution, and thus the evolution is described by the distribution function $p(m^\ell, \sigma^\ell)$. In general, the distribution is described by the three parameters as $p(m^\ell, \sigma^\ell, \eta^\ell)$. In this case, they can be factorized as $p(m^\ell, \sigma^\ell)p(\eta^\ell)$ and thus the evolution of the distribution $p(m^\ell, \sigma^\ell)$ can be described by [YO05]

$$\begin{aligned}
p(m^{\ell+1}, \sigma^{\ell+1}) &= \int dm^\ell \int d\sigma^\ell \int d\eta^\ell p(m^\ell, \sigma^\ell) p(\eta^\ell) \times \\
&\delta(m^{\ell+1} - m^{\ell+1}(m^\ell, \sigma^\ell, \eta^\ell)) \delta(\sigma^{\ell+1} - \sigma^{\ell+1}(m^\ell, \sigma^\ell, \eta^\ell)),
\end{aligned}$$

where $\delta(\cdot)$ is Dirac's delta function and $p(\eta^\ell) = \frac{1}{\sqrt{2\pi}\delta^\ell} \exp\left(-\frac{(\eta^\ell)^2}{2(\delta^\ell)^2}\right)$.

If we set

$$m^{\ell+1}(m^\ell, \sigma^\ell, \eta^\ell) \equiv f(m^\ell, \sigma^\ell, \eta^\ell) = \frac{1}{2} [\text{erf}(u) + \text{erf}(v)], \tag{3.209}$$

$$\sigma^{\ell+1}(m^\ell, \sigma^\ell, \eta^\ell) \equiv g(m^\ell, \sigma^\ell, \eta^\ell) = \sqrt{\alpha + \frac{1}{2\pi}(e^{-u^2} + e^{-v^2})}, \tag{3.210}$$

transforming m^ℓ, σ^ℓ into $f(m^\ell, \sigma^\ell, \eta^\ell), g(m^\ell, \sigma^\ell, \eta^\ell)$, we get the evolution equation of the distribution,

$$p(m^{\ell+1}, \sigma^{\ell+1}) = \int_{-\infty}^{\infty} \frac{d\eta^\ell p(\eta^\ell) p(m^{\ell*}, \sigma^{\ell*})}{\left| \frac{\partial f}{\partial m^\ell} \frac{\partial g}{\partial \sigma^\ell} - \frac{\partial g}{\partial m^\ell} \frac{\partial f}{\partial \sigma^\ell} \right|_{m^\ell=m^{\ell*}, \sigma^\ell=\sigma^{\ell*}}},$$

where $m^{\ell*}, \sigma^{\ell*}$ satisfy eqs.(3.209) and (3.210). Finally we get the probability distribution function [Y005]

$$p(m^{\ell+1}, \sigma^{\ell+1}) = \int_{-\infty}^{\infty} d\eta^{\ell} p(\eta^{\ell}) \frac{2\pi(\eta^{\ell})^2 \sqrt{2\pi\alpha + (e^{-(u^*)^2} + e^{-(v^*)^2})^2}}{(u^* - v^*)^4 e^{-(u^*)^2 - (v^*)^2} (e^{-(u^*)^2} + e^{-(v^*)^2})} p(m^{\ell*}, \sigma^{\ell*}),$$

where $u^* = (m^{\ell*} + \eta^{\ell})/(\sqrt{2}\sigma^{\ell*}), \quad v^* = (m^{\ell*} - \eta^{\ell})/(\sqrt{2}\sigma^{\ell*}).$

3.3 Synergetics of Recurrent and Attractor Neural Nets

Recall that *recurrent neural networks* are neural networks with synaptic feedback loops. Provided that we restrict ourselves to large neural systems, we can apply to their analysis tools from statistical mechanics. Here, we have two possibilities. Under the common conditions of synaptic symmetry, the stochastic process of evolving neuron states leads towards an equilibrium situation where the microscopic state probabilities are known, where the classical techniques of *equilibrium statistical mechanics* can be applied. On the other hand, for non-symmetric networks, where the asymptotic (stationary) statistics are not known, synergetic techniques from *non-equilibrium statistical mechanics* are the only tools available for analysis. Here, the ‘natural’ set of macroscopic *order parameters* to be calculated can be defined in practice as the smallest set which will obey closed deterministic equations in the limit of an infinitely large network.

Being high-dimensional nonlinear systems with extensive feedback, the dynamics of recurrent neural networks are generally dominated by a wealth of different attractors, and the practical use of recurrent neural networks (in both biology and engineering) lies in the potential for creation and manipulation of these attractors through adaptation of the network parameters (synapses and thresholds). Input fed into a recurrent neural network usually serves to induce a specific initial configuration (or firing pattern) of the neurons, which serves as a cue, and the ‘output’ is given by the (static or dynamic) attractor which has been triggered by this cue. The most familiar types of recurrent neural network models, where the idea of creating and manipulating attractors has been worked out and applied explicitly, are the so-called *attractor neural networks* for associative memory, designed to store and retrieve information in the form of neuronal firing patterns and/or sequences of neuronal firing patterns. Each pattern to be stored is represented as a microscopic state vector. One then constructs synapses and thresholds such that the dominant attractors of the network are precisely the pattern vectors (in the case of static recall), or where, alternatively, they are trajectories in which the patterns are successively generated microscopic system states. From an initial configuration (the ‘cue’, or input pattern to be recognized) the system is allowed to

evolve in time autonomously, and the final state (or trajectory) reached can be interpreted as the pattern (or pattern sequence) recognized by network from the input. For such programmes to work one clearly needs recurrent neural networks with extensive *ergodicity breaking*: the state vector will during the course of the dynamics (at least on finite time-scales) have to be confined to a restricted region of state space (an ‘ergodic component’), the location of which is to depend strongly on the initial conditions. Hence our interest will mainly be in systems with many attractors. This, in turn, has implications at a theoretical/mathematical level: solving models of recurrent neural networks with extensively many attractors requires advanced tools from disordered systems theory, such as statical replica theory and dynamical partition function analysis.

The *equilibrium* statistical mechanical techniques can provide much detailed quantitative information on the behavior of recurrent neural networks, but they obviously have serious restrictions. The first one is that, by definition, they will only provide information on network properties in the stationary state. For associative memories, for instance, it is not clear how one can calculate quantities like sizes of domains of attraction without solving the dynamics. The second, and more serious, restriction is that for equilibrium statistical mechanics to apply the dynamics of the network under study must obey detailed balance, i.e., absence of microscopic probability currents in the stationary state. For recurrent networks in which the dynamics take the form of a stochastic alignment of neuronal firing rates to post-synaptic potentials which, in turn, depend linearly on the firing rates, this requirement of detailed balance usually implies symmetry of the synaptic matrix. From a physiological point of view this requirement is clearly unacceptable, since it is violated in any network that obeys Dale’s law as soon as an excitatory neuron is connected to an inhibitory one. Worse still, in any network of graded-response neurons detailed balance will always be violated, even when the synapses are symmetric. The situation will become even worse when we turn to networks of yet more realistic (spike-based) neurons, such as integrate-and-fire ones. In contrast to this, *non-equilibrium* statistical mechanical techniques, it will turn out, do not impose such biologically non-realistic restrictions on neuron types and synaptic symmetry, and they are consequently the more appropriate avenue for future theoretical research aimed at solving biologically more realistic models (for details, see [Coo01, SC00, SC01, CKS05]).

3.3.1 Stochastic Dynamics of Neuronal Firing States

Recall that the simplest non-trivial definition of a recurrent neural network is that where N binary neurons $\sigma_i \in \{-1, 1\}$ (in which the states ‘1’ and ‘-1’ represent firing and rest, respectively) respond iteratively and synchronously to post-synaptic potentials (or local fields) $h_i(\boldsymbol{\sigma})$, with $\boldsymbol{\sigma} = (\sigma_1, \dots, \sigma_N)$. The fields are assumed to depend linearly on the instantaneous neuron states (summation convention upon repeated indices is always used):

$$\text{Parallel Dynamics: } \sigma_i(\ell+1) = \text{sgn}[h_i(\boldsymbol{\sigma}(\ell)) + T\eta_i(\ell)], \quad h_i(\boldsymbol{\sigma}) = J_{ij}\sigma_j + \theta_i. \quad (3.211)$$

The stochasticity is in the independent random numbers $\eta_i(\ell) \in \mathbb{R}$ (representing threshold noise), which are all drawn according to some distribution $w(\eta)$. The parameter T is introduced to control the amount of noise. For $T = 0$ the process (3.211) is deterministic: $\sigma_i(\ell+1) = \text{sgn}[h_i(\boldsymbol{\sigma}(\ell))]$. The opposite extreme is choosing $T = \infty$, here the system evolution is fully random. The external fields θ_i represent neural thresholds and/or external stimuli, J_{ij} represents the synaptic efficacy at the junction $j \rightarrow i$ ($J_{ij} > 0$ implies excitation, $J_{ij} < 0$ inhibition). Alternatively we could decide that at each iteration step ℓ only a single randomly drawn neuron σ_{i_ℓ} is to undergo an update of the type (3.211):

$$\text{Sequential Dynamics: } \begin{cases} i \neq i_\ell : \sigma_i(\ell+1) = \sigma_i(\ell), \\ i = i_\ell : \sigma_i(\ell+1) = \text{sgn}[h_i(\boldsymbol{\sigma}(\ell)) + T\eta_i(\ell)], \end{cases} \quad (3.212)$$

with the local fields as in (3.211). The stochasticity is now both in the independent random numbers $\eta_i(\ell)$ (the threshold noise) and in the site i_ℓ to be updated, drawn randomly from the set $\{1, \dots, N\}$. For simplicity we assume $w(-\eta) = w(\eta)$, and define

$$g(z) = 2 \int_0^z d\eta w(\eta) : \quad g(-z) = -g(z), \quad \lim_{z \rightarrow \pm\infty} g(z) = \pm 1, \quad \partial_z g(z) \geq 0.$$

Popular choices for the threshold noise distributions are

$$\begin{aligned} w(\eta) &= (2\pi)^{-\frac{1}{2}} e^{-\frac{1}{2}\eta^2} : & g(z) &= \text{Erf}[z/\sqrt{2}], \\ w(\eta) &= \frac{1}{2}[1 - \tanh^2(\eta)] : & g(z) &= \tanh(z). \end{aligned}$$

Now, from the microscopic equations (3.211,3.212), which are suitable for numerical simulations, we can derive an equivalent but mathematically more convenient description in terms of microscopic state probabilities $p_\ell(\boldsymbol{\sigma})$. Equations (3.211,3.212) state that, if the system state $\boldsymbol{\sigma}(\ell)$ is given, a neuron i to be updated will obey

$$\text{Prob}[\sigma_i(\ell+1)] = \frac{1}{2} [1 + \sigma_i(\ell+1) g[\beta h_i(\boldsymbol{\sigma}(\ell))]], \quad (3.213)$$

with $\beta = T^{-1}$. In the case (3.211) this rule applies to all neurons, and thus we simply get $p_{\ell+1}(\boldsymbol{\sigma}) = \prod_{i=1}^N \frac{1}{2} [1 + \sigma_i g[\beta h_i(\boldsymbol{\sigma}(\ell))]]$. If, on the other hand, instead of $\boldsymbol{\sigma}(\ell)$ only the probability distribution $p_\ell(\boldsymbol{\sigma})$ is given, this expression for $p_{\ell+1}(\boldsymbol{\sigma})$ is to be averaged over the possible states at time ℓ :

$$\text{Parallel Dynamics : } p_{\ell+1}(\boldsymbol{\sigma}) = \sum_{\boldsymbol{\sigma}'} W[\boldsymbol{\sigma}; \boldsymbol{\sigma}'] p_\ell(\boldsymbol{\sigma}'), \quad (3.214)$$

$$W[\boldsymbol{\sigma}; \boldsymbol{\sigma}'] = \prod_{i=1}^N \frac{1}{2} [1 + \sigma_i g[\beta h_i(\boldsymbol{\sigma}')]].$$

This is the standard representation of a *Markov chain*. Also the sequential process (3.212) can be formulated in terms of probabilities, but here expression (3.213) applies only to the randomly drawn candidate i_ℓ . After averaging over all possible realizations of the sites i_ℓ we get

$$p_{\ell+1}(\boldsymbol{\sigma}) = \frac{1}{N} \left\{ \left[\prod_{j \neq i} \delta_{\sigma_j, \sigma_j(\ell)} \right] \frac{1}{2} [1 + \sigma_i g[\beta h_i(\boldsymbol{\sigma}(\ell))]] \right\},$$

with the Kronecker symbol: $\delta_{ij} = 1$, if $i = j$ and $\delta_{ij} = 0$, otherwise. If, instead of $\boldsymbol{\sigma}(\ell)$, the probabilities $p_\ell(\boldsymbol{\sigma})$ are given, this expression is to be averaged over the possible states at time ℓ , with the result:

$$\begin{aligned} p_{\ell+1}(\boldsymbol{\sigma}) &= \frac{1}{2N} [1 + \sigma_i g[\beta h_i(\boldsymbol{\sigma})]] p_\ell(\boldsymbol{\sigma}) \\ &+ \frac{1}{2N} [1 + \sigma_i g[\beta h_i(F_i \boldsymbol{\sigma})]] p_\ell(F_i \boldsymbol{\sigma}), \end{aligned}$$

with the state-flip operators $F_i \Phi(\boldsymbol{\sigma}) = \Phi(\sigma_1, \dots, \sigma_{i-1}, -\sigma_i, \sigma_{i+1}, \dots, \sigma_N)$. This equation can again be written in the standard form

$$p_{\ell+1}(\boldsymbol{\sigma}) = \sum_{\boldsymbol{\sigma}'} W[\boldsymbol{\sigma}; \boldsymbol{\sigma}'] p_\ell(\boldsymbol{\sigma}'),$$

but now with the transition matrix

$$\text{Sequential Dynamics: } W[\boldsymbol{\sigma}; \boldsymbol{\sigma}'] = \delta_{\boldsymbol{\sigma}, \boldsymbol{\sigma}'} + \frac{1}{N} \{w_i(F_i \boldsymbol{\sigma}) \delta_{\boldsymbol{\sigma}, F_i \boldsymbol{\sigma}'} - w_i(\boldsymbol{\sigma}) \delta_{\boldsymbol{\sigma}, \boldsymbol{\sigma}'}\}, \quad (3.215)$$

$$\text{where } \delta_{\boldsymbol{\sigma}, \boldsymbol{\sigma}'} = \prod_i \delta_{\sigma_i, \sigma'_i} \quad \text{and} \quad w_i(\boldsymbol{\sigma}) = \frac{1}{2} [1 - \sigma_i \tanh[\beta h_i(\boldsymbol{\sigma})]]. \quad (3.216)$$

Note that, as soon as $T > 0$, the two transition matrices $W[\boldsymbol{\sigma}; \boldsymbol{\sigma}']$ in (3.214, 3.215) both describe *ergodic systems*: from any initial state $\boldsymbol{\sigma}'$ one can reach any final state $\boldsymbol{\sigma}$ with nonzero probability in a finite number of steps (being one in the parallel case, and N in the sequential case). It now follows from the standard theory of stochastic processes (see e.g., [Kam92, Gar94]) that in both cases the system evolves towards a unique stationary distribution $p_\infty(\boldsymbol{\sigma})$, where all probabilities $p_\infty(\boldsymbol{\sigma})$ are non-zero [Coo01, SC00, SC01, CKS05].

The above processes have the (mathematically and biologically) less appealing property that time is measured in discrete units. For the sequential case we will now assume that the *duration* of each of the iteration steps is a continuous random number (for parallel dynamics this would make little sense, since all updates would still be made in full synchrony). The statistics of the durations are described by a function $\pi_\ell(t)$, defined as the probability that at time t precisely ℓ updates have been made. Upon denoting the previous discrete-time probabilities as $\hat{p}_\ell(\boldsymbol{\sigma})$, our new process (which now includes the randomness in step duration) will be described by

$$p_t(\boldsymbol{\sigma}) = \sum_{\ell \geq 0} \pi_\ell(t) \hat{p}_\ell(\boldsymbol{\sigma}) = \sum_{\ell \geq 0} \pi_\ell(t) \sum_{\boldsymbol{\sigma}'} W^\ell[\boldsymbol{\sigma}; \boldsymbol{\sigma}'] p_0(\boldsymbol{\sigma}'),$$

and time has become a continuous variable. For $\pi_\ell(t)$ we make the Poisson choice,

$$\pi_\ell(t) = \frac{1}{\ell!} \left(\frac{t}{\Delta}\right)^\ell e^{-t/\Delta}.$$

From $\langle \ell \rangle_\pi = t/\Delta$ and $\langle \ell^2 \rangle_\pi = t/\Delta + t^2/\Delta^2$ it follows that Δ is the average duration of an iteration step, and that the relative deviation in ℓ at a given t vanishes for $\Delta \rightarrow 0$ as $\sqrt{\langle \ell^2 \rangle_\pi - \langle \ell \rangle_\pi^2} / \langle \ell \rangle_\pi = \sqrt{\Delta/t}$. The nice properties of the Poisson distribution under temporal derivation allow us to derive:

$$\Delta \dot{p}_t(\boldsymbol{\sigma}) = \sum_{\boldsymbol{\sigma}'} W[\boldsymbol{\sigma}; \boldsymbol{\sigma}'] p_t(\boldsymbol{\sigma}') - p_t(\boldsymbol{\sigma}).$$

For sequential dynamics, we choose $\Delta = \frac{1}{N}$ so that, as in the parallel case, in one time unit each neuron will on average be updated once. The master equation corresponding to (3.215) acquires the form $w_i(\boldsymbol{\sigma})$; (3.216) now play the role of *transition rates*. The choice $\Delta = \frac{1}{N}$ implies $\sqrt{\langle \ell^2 \rangle_\pi - \langle \ell \rangle_\pi^2} / \langle \ell \rangle_\pi = \sqrt{1/Nt}$, so we will still for $N \rightarrow \infty$ no longer have uncertainty in where we are on the t axis.

Alternatively, we could start with continuous neuronal variables σ_i (representing e.g., firing frequencies or oscillator phases), where $i = 1, \dots, N$, and with stochastic equations of the form

$$\sigma_i(t + \Delta) = \sigma_i(t) + \Delta f_i(\boldsymbol{\sigma}(t)) + \sqrt{2T\Delta} \xi_i(t). \quad (3.217)$$

Here, we have introduced (as yet unspecified) deterministic state-dependent forces $f_i(\boldsymbol{\sigma})$, and uncorrelated Gaussian distributed random forces $\xi_i(t)$ (the noise), with

$$\langle \xi_i(t) \rangle = 0 \quad \text{and} \quad \langle \xi_i(t) \xi_j(t') \rangle = \delta_{ij} \delta_{t,t'}.$$

As before, the parameter T controls the amount of noise in the system, ranging from $T = 0$ (deterministic dynamics) to $T = \infty$ (completely random dynamics). If we take the limit $\Delta \rightarrow 0$ in (3.217) we find a Langevin equation (with a continuous time variable) [Coo01, SC00, SC01, CKS05]:

$$\dot{\sigma}_i(t) = f_i(\boldsymbol{\sigma}(t)) + \eta_i(t). \quad (3.218)$$

This equation acquires its meaning only as the limit $\Delta \rightarrow 0$ of (3.217). The moments of the new noise variables $\eta_i(t) = \xi_i(t) \sqrt{2T/\Delta}$ in (3.218) are given by

$$\langle \eta_i(t) \rangle = 0 \quad \text{and} \quad \langle \eta_i(t) \eta_j(t') \rangle = 2T \delta_{ij} \delta(t - t').$$

This can be derived from the moments of the $\xi_i(t)$. For instance:

$$\begin{aligned}\langle \eta_i(t) \eta_j(t') \rangle &= \lim_{\Delta \rightarrow 0} \frac{2T}{\Delta} \langle \xi_i(t) \xi_j(t') \rangle \\ &= 2T \delta_{ij} \lim_{\Delta \rightarrow 0} \frac{1}{\Delta} \delta_{t,t'} = 2TC \delta_{ij} \delta(t-t').\end{aligned}$$

The constant C is found by summing over t' , before taking the limit $\Delta \rightarrow 0$, in the above equation:

$$\begin{aligned}\int dt' \langle \eta_i(t) \eta_j(t') \rangle &= \lim_{\Delta \rightarrow 0} 2T \sum_{t'=-\infty}^{\infty} \langle \xi_i(t) \xi_j(t') \rangle \\ &= 2T \delta_{ij} \lim_{\Delta \rightarrow 0} \sum_{t'=-\infty}^{\infty} \delta_{t,t'} = 2T \delta_{ij}.\end{aligned}$$

Thus $C = 1$, which indeed implies $\langle \eta_i(t) \eta_j(t') \rangle = 2T \delta_{ij} \delta(t-t')$. More directly, one can also calculate the moment partition function:

$$\langle e^{i \int dt \psi_i(t) \eta_i(t)} \rangle = \lim_{\Delta \rightarrow 0} \prod_{i,t} \int \frac{dz}{\sqrt{2\pi}} e^{-\frac{1}{2} z^2 + i z \psi_i(t) \sqrt{2T\Delta}} \quad (3.219)$$

$$= \lim_{\Delta \rightarrow 0} \prod_{i,t} e^{-T \Delta \psi_i^2(t)} = e^{-T \int dt \sum_i \psi_i^2(t)}. \quad (3.220)$$

On the other hand, a mathematically more convenient description of the process (3.218) is provided by the Fokker–Planck equation for the microscopic state probability density $p_t(\boldsymbol{\sigma}) = \langle \delta[\boldsymbol{\sigma} - \boldsymbol{\sigma}(t)] \rangle$, which we will now derive. For the discrete-time process (3.217) we expand the δ -distribution in the definition of $p_{t+\Delta}(\boldsymbol{\sigma})$ (in a distributional sense) [Coo01, SC00, SC01, CKS05]:

$$\begin{aligned}p_{t+\Delta}(\boldsymbol{\sigma}) - p_t(\boldsymbol{\sigma}) &= \langle \delta[\boldsymbol{\sigma} - \boldsymbol{\sigma}(t) - \Delta \mathbf{f}(\boldsymbol{\sigma}(t)) - \sqrt{2T\Delta} \boldsymbol{\xi}(t)] \rangle - \langle \delta[\boldsymbol{\sigma} - \boldsymbol{\sigma}(t)] \rangle \\ &= -\frac{\partial}{\partial \sigma_i} \langle \delta[\boldsymbol{\sigma} - \boldsymbol{\sigma}(t)] [\Delta f_i(\boldsymbol{\sigma}(t)) + \sqrt{2T\Delta} \xi_i(t)] \rangle \\ &\quad + T\Delta \frac{\partial^2}{\partial \sigma_i \partial \sigma_j} \langle \delta[\boldsymbol{\sigma} - \boldsymbol{\sigma}(t)] \xi_i(t) \xi_j(t) \rangle + \mathcal{O}(\Delta^{\frac{3}{2}}).\end{aligned}$$

The variables $\boldsymbol{\sigma}(t)$ depend only on noise variables $\xi_j(t')$ with $t' < t$, so that for any function A ,

$$\begin{aligned}\langle A[\boldsymbol{\sigma}(t)] \xi_i(t) \rangle &= \langle A[\boldsymbol{\sigma}(t)] \rangle \langle \xi_i(t) \rangle = 0, \quad \text{and} \\ \langle A[\boldsymbol{\sigma}(t)] \xi_i(t) \xi_j(t) \rangle &= \delta_{ij} \langle A[\boldsymbol{\sigma}(t)] \rangle.\end{aligned}$$

As a consequence, we have:

$$\begin{aligned}\frac{1}{\Delta} [p_{t+\Delta}(\boldsymbol{\sigma}) - p_t(\boldsymbol{\sigma})] &= -\frac{\partial}{\partial \sigma_i} \langle \delta[\boldsymbol{\sigma} - \boldsymbol{\sigma}(t)] f_i(\boldsymbol{\sigma}(t)) \rangle \\ &\quad + T \frac{\partial^2}{\partial \sigma_i^2} \langle \delta[\boldsymbol{\sigma} - \boldsymbol{\sigma}(t)] \rangle + \mathcal{O}(\Delta^{\frac{1}{2}}) \\ &= -\frac{\partial}{\partial \sigma_i} [p_t(\boldsymbol{\sigma}) f_i(\boldsymbol{\sigma})] + T \frac{\partial^2}{\partial \sigma_i^2} p_t(\boldsymbol{\sigma}) + \mathcal{O}(\Delta^{\frac{1}{2}}).\end{aligned}$$

By taking the limit $\Delta \rightarrow 0$ we then arrive at the Fokker-Planck equation:

$$\dot{p}_t(\boldsymbol{\sigma}) = -\frac{\partial}{\partial \sigma_i} [p_t(\boldsymbol{\sigma}) f_i(\boldsymbol{\sigma})] + T \frac{\partial^2}{\partial \sigma_i^2} p_t(\boldsymbol{\sigma}). \quad (3.221)$$

In the case of graded-response neurons, the continuous variable σ_i represents the membrane potential of neuron i , and (in their simplest form) the deterministic forces are given by

$$f_i(\boldsymbol{\sigma}) = J_{ij} \tanh[\gamma \sigma_j] - \sigma_i + \theta_i, \quad \text{with} \quad \gamma > 0,$$

and with the θ_i representing injected currents. Conventional notation is restored by putting $\sigma_i \rightarrow u_i$. Thus equation (3.218) specializes to

$$\dot{u}_i(t) = J_{ij} \tanh[\gamma u_j(t)] - u_i(t) + \theta_i + \eta_i(t). \quad (3.222)$$

One often chooses $T = 0$ (i.e., $\eta_i(t) = 0$), the rationale being that threshold noise is already assumed to have been incorporated via the nonlinearity in (3.222).

In our second example the variables σ_i represent the phases of coupled neural oscillators, with forces of the form

$$f_i(\boldsymbol{\sigma}) = J_{ij} \sin(\sigma_j - \sigma_i) + \omega_i.$$

Individual synapses J_{ij} now try to enforce either pair-wise synchronization ($J_{ij} > 0$), or pair-wise anti-synchronization ($J_{ij} < 0$), and the ω_i represent the natural frequencies of the individual oscillators. Conventional notation dictates $\sigma_i \rightarrow \xi_i$, giving [Coo01, SC00, SC01, CKS05]

$$\dot{\xi}_i(t) = \omega_i + J_{ij} \sin[\xi_j(t) - \xi_i(t)] + \eta_i(t). \quad (3.223)$$

3.3.2 Synaptic Symmetry and Lyapunov Functions

In the deterministic limit $T \rightarrow 0$ the rules (3.211) for networks of synchronously evolving binary neurons reduce to the deterministic map

$$\sigma_i(\ell + 1) = \text{sgn}[h_i(\boldsymbol{\sigma}(\ell))]. \quad (3.224)$$

It turns out that for systems with symmetric interactions, $J_{ij} = J_{ji}$ for all (ij) , one can construct a Lyapunov function, i.e., a function of $\boldsymbol{\sigma}$ which during the dynamics decreases monotonically and is bounded from below (see e.g., [Kha92]):

$$\text{Binary \& Parallel Dynamics:} \quad L[\boldsymbol{\sigma}] = -\sum_i |h_i(\boldsymbol{\sigma})| - \sigma_i \theta_i. \quad (3.225)$$

Clearly, $L \geq -\sum_i [\sum_j |J_{ij}| + |\theta_i|] - \sum_i |\theta_i|$. During iteration of (3.224) we find:

$$\begin{aligned}
L[\boldsymbol{\sigma}(\ell+1)] - L[\boldsymbol{\sigma}(\ell)] &= - \sum_i |h_i(\boldsymbol{\sigma}(\ell+1))| \\
&\quad + \sigma_i(\ell+1)[J_{ij}\sigma_j(\ell) + \theta_i] - \theta_i [\sigma_i(\ell+1) - \sigma_i(\ell)] \\
&= - \sum_i |h_i(\boldsymbol{\sigma}(\ell+1))| + \sigma_i(\ell)h_i(\boldsymbol{\sigma}(\ell+1)) \\
&= - \sum_i |h_i(\boldsymbol{\sigma}(\ell+1))| [1 - \sigma_i(\ell+2)\sigma_i(\ell)] \leq 0,
\end{aligned}$$

where we have used (3.224) and $J_{ij} = J_{ji}$. So L decreases monotonically until a stage is reached where $\sigma_i(\ell+2) = \sigma_i(\ell)$ for all i . Thus, with symmetric interactions this system will in the deterministic limit always end up in a limit cycle with period ≤ 2 . A similar result is found for networks with binary neurons and sequential dynamics. In the limit $T \rightarrow 0$ the rules (3.212) reduce to the map

$$\sigma_i(\ell+1) = \delta_{i,i_\ell} \text{sgn}[h_i(\boldsymbol{\sigma}(\ell))] + [1 - \delta_{i,i_\ell}] \sigma_i(\ell). \quad (3.226)$$

(in which we still have randomness in the choice of site to be updated). For systems with symmetric interactions and without self-interactions, i.e., $J_{ii} = 0$ for all i , we again find a Lyapunov function:

$$\text{Binary \& Sequential Dynamics:} \quad L[\boldsymbol{\sigma}] = -\frac{1}{2} \sigma_i J_{ij} \sigma_j - \sigma_i \theta_i. \quad (3.227)$$

This quantity is bounded from below, $L \geq -\frac{1}{2} \sum_{ij} |J_{ij}| - \sum_i |\theta_i|$. Upon calling the site i_ℓ selected for update at step ℓ simply i , the change in L during iteration of (3.226) can be written as [Coo01, SC00, SC01, CKS05]:

$$\begin{aligned}
L[\boldsymbol{\sigma}(\ell+1)] - L[\boldsymbol{\sigma}(\ell)] &= -\theta_i [\sigma_i(\ell+1) - \sigma_i(\ell)] \\
&\quad - \frac{1}{2} J_{ik} [\sigma_i(\ell+1) \sigma_k(\ell+1) - \sigma_i(\ell) \sigma_k(\ell)] \\
&\quad - \frac{1}{2} J_{ji} [\sigma_j(\ell+1) \sigma_i(\ell+1) - \sigma_j(\ell) \sigma_i(\ell)] \\
&= [\sigma_i(\ell) - \sigma_i(\ell+1)] [J_{ij} \sigma_j(\ell) + \theta_i] \\
&= -|h_i(\boldsymbol{\sigma}(\ell))| [1 - \sigma_i(\ell) \sigma_i(\ell+1)] \leq 0.
\end{aligned}$$

Here we used (3.226), $J_{ij} = J_{ji}$, and absence of self-interactions. Thus L decreases monotonically until $\sigma_i(t+1) = \sigma_i(t)$ for all i . With symmetric synapses, but without diagonal terms, the sequentially evolving binary neurons system will in the deterministic limit always end up in a stationary state.

Now, one can derive similar results for models with continuous variables. Firstly, in the deterministic limit the graded-response equations (3.222) simplify to

$$\dot{u}_i(t) = J_{ij} \tanh[\gamma u_j(t)] - u_i(t) + \theta_i. \quad (3.228)$$

Symmetric networks again admit a Lyapunov function (without a need to eliminate self-interactions):

Graded-Response Dynamics :
$$L[\mathbf{u}] = -\frac{1}{2} J_{ij} \tanh(\gamma u_i) \tanh(\gamma u_j) + \sum_i \left[\gamma \int_0^{u_i} dv v [1 - \tanh^2(\gamma v)] - \theta_i \tanh(\gamma u_i) \right].$$

Clearly, $L \geq -\frac{1}{2} \sum_{ij} |J_{ij}| - \sum_i |\theta_i|$; the term in $L[\mathbf{u}]$ with the integral is non-negative. During the noise-free dynamics (3.228) one can use the identity

$$\frac{\partial L}{\partial u_i} = -\gamma \dot{u}_i [1 - \tanh^2(\gamma u_i)],$$

valid only when $J_{ij} = J_{ji}$, to derive [Coo01, SC00, SC01, CKS05]

$$\dot{L} = \frac{\partial L}{\partial u_i} \dot{u}_i = -\gamma \sum_i [1 - \tanh^2(\gamma u_i)] \dot{u}_i^2 \leq 0.$$

Again L is found to decrease monotonically, until $\dot{u}_i = 0$ for all i , i.e., until we are at a fixed-point.

The coupled oscillator equations (3.223) reduce in the *noise-free limit* to

$$\dot{\xi}_i(t) = \omega_i + J_{ij} \sin[\xi_j(t) - \xi_i(t)]. \quad (3.229)$$

Note that self-interactions J_{ii} always drop out automatically. For symmetric oscillator networks, a construction of the type followed for the graded-response equations would lead us to propose

Coupled Oscillators Dynamics:
$$L[\boldsymbol{\xi}] = -\frac{1}{2} J_{ij} \cos[\xi_i - \xi_j] - \omega_i \xi_i. \quad (3.230)$$

This function decreases monotonically, due to $\partial L / \partial \xi_i = -\dot{\xi}_i$:

$$\dot{L} = \frac{\partial L}{\partial \xi_i} \dot{\xi}_i = -\sum_i \dot{\xi}_i^2 \leq 0.$$

Actually, (3.229) describes gradient descent on the surface $L[\boldsymbol{\xi}]$. However, due to the term with the natural frequencies ω_i the function $L[\boldsymbol{\xi}]$ is not bounded, so it cannot be a Lyapunov function. This could have been expected; when $J_{ij} = 0$ for all (i, j) , for instance, one finds continually increasing phases, $\xi_i(t) = \xi_i(0) + \omega_i t$. Removing the ω_i , in contrast, gives the bound $L \geq -\sum_j |J_{ij}|$. Now the system must go to a fixed-point. In the special case $\omega_i = \omega$ (N identical natural frequencies) we can transform away the ω_i by putting $\xi(t) = \tilde{\xi}_i(t) + \omega t$, and find the relative phases $\tilde{\xi}_i$ to go to a fixed-point.

3.3.3 Balance and Equilibrium Statistical Mechanics

The results got above indicate that networks with symmetric synapses are a special class [Coo01, SC00, SC01, CKS05]. We now show how synaptic

symmetry is closely related to the detailed balance property, and derive a number of consequences. An ergodic Markov chain of the form (3.214,3.215), i.e.,

$$p_{\ell+1}(\sigma) = \sum_{\sigma'} W[\sigma; \sigma'] p_{\ell}(\sigma'), \quad (3.231)$$

is said to obey detailed balance if its (unique) stationary solution $p_{\infty}(\sigma)$ has the property

$$W[\sigma; \sigma'] p_{\infty}(\sigma') = W[\sigma'; \sigma] p_{\infty}(\sigma), \quad (\text{for all } \sigma, \sigma'). \quad (3.232)$$

All $p_{\infty}(\sigma)$ which satisfy (3.232) are stationary solutions of (3.231), this is easily verified by substitution. The converse is not true. Detailed balance states that, in addition to $p_{\infty}(\sigma)$ being stationary, one has *equilibrium*: there is no net probability current between any two microscopic system states.

It is not a trivial matter to investigate systematically for which choices of the threshold noise distribution $w(\eta)$ and the synaptic matrix $\{J_{ij}\}$ detailed balance holds. It can be shown that, apart from trivial cases (e.g., systems with self-interactions only) a Gaussian distribution $w(\eta)$ will not support detailed balance. Here we will work out details only for the choice $w(\eta) = \frac{1}{2}[1 - \tanh^2(\eta)]$, and for $T > 0$ (where both discrete systems are ergodic). For parallel dynamics the transition matrix is given in (3.214), now with $g[z] = \tanh[z]$, and the detailed balance condition (3.232) becomes

$$\frac{e^{\beta \sigma_i h_i(\sigma')} p_{\infty}(\sigma')}{\prod_i \cosh[\beta h_i(\sigma')]} = \frac{e^{\beta \sigma'_i h_i(\sigma)} p_{\infty}(\sigma)}{\prod_i \cosh[\beta h_i(\sigma)]}, \quad (\text{for all } \sigma, \sigma'). \quad (3.233)$$

All $p_{\infty}(\sigma)$ are non-zero (ergodicity), so we may safely put

$$p_{\infty}(\sigma) = e^{\beta[\theta_i \sigma_i + K(\sigma)]} \prod_i \cosh[\beta h_i(\sigma)],$$

which, in combination with definition (3.211), simplifies the detailed balance condition to:

$$K(\sigma) - K(\sigma') = \sigma_i [J_{ij} - J_{ji}] \sigma'_j, \quad (\text{for all } \sigma, \sigma'). \quad (3.234)$$

Averaging (3.234) over all possible σ' gives $K(\sigma) = \langle K(\sigma') \rangle_{\sigma'}$ for all σ , i.e., K is a constant, whose value follows from normalizing $p_{\infty}(\sigma)$. So, if detailed balance holds the equilibrium distribution must be [Coo01, SC00, SC01, CKS05]:

$$p_{\text{eq}}(\sigma) \sim e^{\beta \theta_i \sigma_i} \prod_i \cosh[\beta h_i(\sigma)]. \quad (3.235)$$

For symmetric systems detailed balance indeed holds: (3.235) solves (3.233), since $K(\sigma) = K$ solves the reduced problem (3.234). For non-symmetric systems, however, there can be no equilibrium. For $K(\sigma) = K$ the condition (3.234) becomes $\sum_{ij} \sigma_i [J_{ij} - J_{ji}] \sigma'_j = 0$ for all $\sigma, \sigma' \in \{-1, 1\}^N$. For $N \geq 2$

the vector pairs $(\boldsymbol{\sigma}, \boldsymbol{\sigma}')$ span the space of all $N \times N$ matrices, so $J_{ij} - J_{ji}$ must be zero. For $N = 1$ there simply exists no non-symmetric synaptic matrix. In conclusion: for binary networks with parallel dynamics, interaction symmetry implies detailed balance, and vice versa.

For sequential dynamics, with $w(\eta) = \frac{1}{2}[1 - \tanh^2(\eta)]$, the transition matrix is given by (3.215) and the detailed balance condition (3.232) simplifies to

$$\frac{e^{\beta \sigma_i h_i(F_i \boldsymbol{\sigma})} p_\infty(F_i \boldsymbol{\sigma})}{\cosh[\beta h_i(F_i \boldsymbol{\sigma})]} = \frac{e^{-\beta \sigma_i h_i(\boldsymbol{\sigma})} p_\infty(\boldsymbol{\sigma})}{\cosh[\beta h_i(\boldsymbol{\sigma})]}, \quad (\text{for all } \boldsymbol{\sigma}, i).$$

Self-interactions J_{ii} , inducing $h_i(F_i \boldsymbol{\sigma}) \neq h_i(\boldsymbol{\sigma})$, complicate matters. Therefore we first consider systems where all $J_{ii} = 0$. All stationary probabilities $p_\infty(\boldsymbol{\sigma})$ being non-zero (ergodicity), we may write:

$$p_\infty(\boldsymbol{\sigma}) = e^{\beta[\theta_i \sigma_i + \frac{1}{2} \sigma_i J_{ij} \sigma_j + K(\boldsymbol{\sigma})]}. \quad (3.236)$$

Using relations like

$$J_{kl} F_i(\sigma_k \sigma_l) = J_{kl} \sigma_k \sigma_l - 2\sigma_i [J_{ik} + J_{ki}] \sigma_k,$$

we can simplify the detailed balance condition to

$$K(F_i \boldsymbol{\sigma}) - K(\boldsymbol{\sigma}) = \sigma_i [J_{ik} - J_{ki}] \sigma_k, \quad (\text{for all } \boldsymbol{\sigma}, i).$$

If to this expression we apply the general identity

$$[1 - F_i] f(\boldsymbol{\sigma}) = 2\sigma_i \langle \sigma_i f(\boldsymbol{\sigma}) \rangle_{\sigma_i},$$

we find for $i \neq j$ [Coo01, SC00, SC01, CKS05]:

$$K(\boldsymbol{\sigma}) = -2\sigma_i \sigma_j [J_{ij} - J_{ji}], \quad (\text{for all } \boldsymbol{\sigma} \text{ and all } i \neq j).$$

The left-hand side is symmetric under permutation of the pair (i, j) , which implies that the interaction matrix must also be symmetric: $J_{ij} = J_{ji}$ for all (i, j) . We now find the trivial solution $K(\boldsymbol{\sigma}) = K$ (constant), detailed balance holds and the corresponding equilibrium distribution is

$$p_{\text{eq}}(\boldsymbol{\sigma}) \sim e^{-\beta H(\boldsymbol{\sigma})}, \quad H(\boldsymbol{\sigma}) = -\frac{1}{2} \sigma_i J_{ij} \sigma_j - \theta_i \sigma_i.$$

In conclusion: for binary networks with sequential dynamics, but without self-interactions, interaction symmetry implies detailed balance, and vice versa. In the case of self-interactions the situation is more complicated. However, here one can still show that non-symmetric models with detailed balance must be pathological, since the requirements can be met only for very specific choices for the $\{J_{ij}\}$.

Now, let us turn to the question of when we find microscopic equilibrium (stationarity without probability currents) in continuous models described

by a Fokker–Planck equation (3.221). Note that (3.221) can be seen as a continuity equation for the density of a conserved quantity:

$$\dot{p}_t(\boldsymbol{\sigma}) + \frac{\partial}{\partial \sigma_i} J_i(\boldsymbol{\sigma}, t) = 0.$$

The components $J_i(\boldsymbol{\sigma}, t)$ of the current density are given by

$$J_i(\boldsymbol{\sigma}, t) = [f_i(\boldsymbol{\sigma}) - T \frac{\partial}{\partial \sigma_i}] p_t(\boldsymbol{\sigma}).$$

Stationary distributions $p_\infty(\boldsymbol{\sigma})$ are those which give $\sum_i \frac{\partial}{\partial \sigma_i} J_i(\boldsymbol{\sigma}, \infty) = 0$ (divergence-free currents). Detailed balance implies the stronger statement $J_i(\boldsymbol{\sigma}, \infty) = 0$ for all i (zero currents), so

$$f_i(\boldsymbol{\sigma}) = T \frac{\partial \log p_\infty(\boldsymbol{\sigma})}{\partial \sigma_i}, \quad \text{or}$$

$$f_i(\boldsymbol{\sigma}) = -\frac{\partial H(\boldsymbol{\sigma})}{\partial \sigma_i}, \quad p_\infty(\boldsymbol{\sigma}) \sim e^{-\beta H(\boldsymbol{\sigma})}, \quad (3.237)$$

for some $H(\boldsymbol{\sigma})$, i.e., the forces $f_i(\boldsymbol{\sigma})$ must be conservative. However, one can have conservative forces without a normalizable equilibrium distribution. Just take $H(\boldsymbol{\sigma}) = 0$, i.e., $f_i(\boldsymbol{\sigma}, t) = 0$: here we have $p_{\text{eq}}(\boldsymbol{\sigma}) = C$, which is not normalizable for $\boldsymbol{\sigma} \in \mathbb{R}^N$. For this particular case equation (3.221) is solved easily:

$$p_t(\boldsymbol{\sigma}) = [4\pi T t]^{-N/2} \int d\boldsymbol{\sigma}' p_0(\boldsymbol{\sigma}') e^{-[\boldsymbol{\sigma} - \boldsymbol{\sigma}']^2 / 4T t},$$

so the limit $\lim_{t \rightarrow \infty} p_t(\boldsymbol{\sigma})$ does not exist. One can prove the following (see e.g., [Zin93]). If the forces are conservative and if $p_\infty(\boldsymbol{\sigma}) \sim e^{-\beta H(\boldsymbol{\sigma})}$ is normalizable, then it is the unique stationary solution of the Fokker–Planck equation, to which the system converges for all initial distributions $p_0 \in L^1[\mathbb{R}^N]$ which obey $\int_{\mathbb{R}^N} d\boldsymbol{\sigma} e^{\beta H(\boldsymbol{\sigma})} p_0^2(\boldsymbol{\sigma}) < \infty$.

Note that conservative forces must obey [Coo01, SC00, SC01, CKS05]

$$\frac{\partial f_i(\boldsymbol{\sigma})}{\partial \sigma_j} - \frac{\partial f_j(\boldsymbol{\sigma})}{\partial \sigma_i} = 0, \quad (\text{for all } \boldsymbol{\sigma} \text{ and all } i \neq j). \quad (3.238)$$

In the graded–response equations (3.228) the deterministic forces are

$$f_i(\mathbf{u}) = J_{ij} \tanh[\gamma u_j] - u_i + \theta_i, \quad \text{where} \\ \frac{\partial f_i(\mathbf{u})}{\partial u_j} - \frac{\partial f_j(\mathbf{u})}{\partial u_i} = \gamma \{J_{ij} [1 - \tanh^2[\gamma u_j]] - J_{ji} [1 - \tanh^2[\gamma u_i]]\}.$$

At $\mathbf{u} = \mathbf{0}$ this reduces to $J_{ij} - J_{ji}$, i.e., the interaction matrix must be symmetric. For symmetric matrices we find away from $\mathbf{u} = \mathbf{0}$:

$$\frac{\partial f_i(\mathbf{u})}{\partial u_j} - \frac{\partial f_j(\mathbf{u})}{\partial u_i} = \gamma J_{ij} \{ \tanh^2[\gamma u_i] - \tanh^2[\gamma u_j] \}.$$

The only way for this to be zero for any $\mathbf{u}$ is by having $J_{ij} = 0$ for all $i \neq j$, i.e., all neurons are disconnected (in this trivial case the system (3.228) does indeed obey detailed balance). Network models of interacting graded-response neurons of the type (3.228) apparently never reach equilibrium, they will always violate detailed balance and exhibit microscopic probability currents. In the case of coupled oscillators (3.223), where the deterministic forces are

$$f_i(\boldsymbol{\xi}) = J_{ij} \sin[\xi_j - \xi_i] + \omega_i,$$

one finds the left-hand side of condition (3.238) to give

$$\frac{\partial f_i(\boldsymbol{\xi})}{\partial \xi_j} - \frac{\partial f_j(\boldsymbol{\xi})}{\partial \xi_i} = [J_{ij} - J_{ji}] \cos[\xi_j - \xi_i].$$

Requiring this to be zero for any $\boldsymbol{\xi}$ gives the condition $J_{ij} = J_{ji}$ for any $i \neq j$. We have already seen that symmetric oscillator networks indeed have conservative forces:

$$f_i(\boldsymbol{\xi}) = -\partial H(\boldsymbol{\xi})/\partial \xi_i, \quad \text{with} \quad H(\boldsymbol{\xi}) = -\frac{1}{2} J_{ij} \cos[\xi_i - \xi_j] - \omega_i \xi_i.$$

If in addition we choose all $\omega_i = 0$ the function $H(\boldsymbol{\sigma})$ will also be bounded from below, and, although $p_\infty(\boldsymbol{\xi}) \sim e^{-\beta H(\boldsymbol{\xi})}$ is still not normalizable on $\boldsymbol{\xi} \in \mathbb{R}^N$, the full 2π -periodicity of the function $H(\boldsymbol{\sigma})$ now allows us to identify $\xi_i + 2\pi \equiv \xi_i$ for all i , so that now $\boldsymbol{\xi} \in [-\pi, \pi]^N$ and $\int d\boldsymbol{\xi} e^{-\beta H(\boldsymbol{\xi})}$ does exist. Thus symmetric coupled oscillator networks with zero natural frequencies obey detailed balance. In the case of non-zero natural frequencies, in contrast, detailed balance does not hold.

The above results establish the link with *equilibrium statistical mechanics* (see e.g., [Yeo92, PB94]). For binary systems with symmetric synapses (in the sequential case: without self-interactions) and with threshold noise distributions of the form

$$w(\eta) = \frac{1}{2}[1 - \tanh^2(\eta)],$$

detailed balance holds and we know the equilibrium distributions. For sequential dynamics it has the Boltzmann form (3.237) and we can apply standard equilibrium statistical mechanics. The parameter β can formally be identified with the inverse ‘temperature’ in equilibrium, $\beta = T^{-1}$, and the function $H(\boldsymbol{\sigma})$ is the usual *Ising-spin Hamiltonian*. In particular we can define the *partition function* Z and the *free energy* F [Coo01, SC00, SC01, CKS05]:

$$p_{\text{eq}}(\boldsymbol{\sigma}) = \frac{1}{Z} e^{-\beta H(\boldsymbol{\sigma})}, \quad H(\boldsymbol{\sigma}) = -\frac{1}{2} \sigma_i J_{ij} \sigma_j - \theta_i \sigma_i, \quad (3.239)$$

$$Z = \sum_{\boldsymbol{\sigma}} e^{-\beta H(\boldsymbol{\sigma})}, \quad F = -\beta^{-1} \log Z. \quad (3.240)$$

The free energy can be used as the partition function for equilibrium averages. Taking derivatives with respect to external fields θ_i and interactions J_{ij} , for

instance, produces $\langle \sigma_i \rangle = -\partial F / \partial \theta_i$ and $\langle \sigma_i \sigma_j \rangle = -\partial F / \partial J_{ij}$, whereas equilibrium averages of arbitrary state variable $f(\boldsymbol{\sigma})$ can be obtained by adding suitable partition terms to the Hamiltonian:

$$H(\boldsymbol{\sigma}) \rightarrow H(\boldsymbol{\sigma}) + \lambda f(\boldsymbol{\sigma}), \quad \langle f \rangle = \lim_{\lambda \rightarrow 0} \frac{\partial F}{\partial \lambda}.$$

In the parallel case (3.235) we can again formally write the equilibrium probability distribution in the Boltzmann form [Per84] and define a corresponding partition function $\tilde{Z}$ and a free energy $\tilde{F}$:

$$p_{\text{eq}}(\boldsymbol{\sigma}) = \frac{1}{\tilde{Z}} e^{-\beta \tilde{H}(\boldsymbol{\sigma})}, \quad \tilde{H}(\boldsymbol{\sigma}) = -\theta_i \sigma_i - \frac{1}{\beta} \sum_i \log 2 \cosh[\beta h_i(\boldsymbol{\sigma})], \quad (3.241)$$

$$\tilde{Z} = \sum_{\boldsymbol{\sigma}} e^{-\beta \tilde{H}(\boldsymbol{\sigma})}, \quad \tilde{F} = -\beta^{-1} \log \tilde{Z}, \quad (3.242)$$

which again serve to generate averages: $\tilde{H}(\boldsymbol{\sigma}) \rightarrow \tilde{H}(\boldsymbol{\sigma}) + \lambda f(\boldsymbol{\sigma})$, $\langle f \rangle = \lim_{\lambda \rightarrow 0} \partial \tilde{F} / \partial \lambda$. However, standard thermodynamic relations involving derivation with respect to β need no longer be valid, and derivation with respect to fields or interactions generates different types of averages, such as [Coo01, SC00, SC01, CKS05]

$$\begin{aligned} -\frac{\partial \tilde{F}}{\partial \theta_i} &= \langle \sigma_i \rangle + \langle \tanh[\beta h_i(\boldsymbol{\sigma})] \rangle, & -\frac{\partial \tilde{F}}{\partial J_{ii}} &= \langle \sigma_i \tanh[\beta h_i(\boldsymbol{\sigma})] \rangle, \\ i \neq j: & \frac{\partial \tilde{F}}{\partial J_{ij}} &= \langle \sigma_i \tanh[\beta h_j(\boldsymbol{\sigma})] \rangle + \langle \sigma_j \tanh[\beta h_i(\boldsymbol{\sigma})] \rangle. \end{aligned}$$

One can use $\langle \sigma_i \rangle = \langle \tanh[\beta h_i(\boldsymbol{\sigma})] \rangle$, which can be derived directly from the equilibrium equation $p_{\text{eq}}(\boldsymbol{\sigma}) = \sum_{\boldsymbol{\sigma}'} W[\boldsymbol{\sigma}; \boldsymbol{\sigma}'] p_{\text{eq}}(\boldsymbol{\sigma}')$, to simplify the first of these identities.

A connected network of graded-response neurons can never be in an equilibrium state, so our only model example with continuous neuronal variables for which we can set up the equilibrium statistical mechanics formalism is the system of coupled oscillators (3.223) with symmetric synapses and absent (or uniform) natural frequencies ω_i . If we define the phases as $\xi_i \in [-\pi, \pi]$ we have again an equilibrium distribution of the Boltzmann form, and we can define the standard thermodynamic quantities:

$$p_{\text{eq}}(\boldsymbol{\xi}) = \frac{1}{Z} e^{-\beta H(\boldsymbol{\xi})}, \quad H(\boldsymbol{\xi}) = -\frac{1}{2} J_{ij} \cos[\xi_i - \xi_j], \quad (3.243)$$

$$Z = \int_{-\pi}^{\pi} \dots \int_{-\pi}^{\pi} d\boldsymbol{\xi} e^{-\beta H(\boldsymbol{\xi})}, \quad F = -\beta^{-1} \log Z. \quad (3.244)$$

These generate equilibrium averages in the usual manner. For instance

$$\langle \cos[\xi_i - \xi_j] \rangle = -\frac{\partial F}{\partial J_{ij}},$$

whereas averages of arbitrary state variables $f(\boldsymbol{\xi})$ follow, as before, upon introducing suitable partition terms:

$$H(\boldsymbol{\xi}) \rightarrow H(\boldsymbol{\xi}) + \lambda f(\boldsymbol{\xi}), \quad \langle f \rangle = \lim_{\lambda \rightarrow 0} \frac{\partial F}{\partial \lambda}.$$

3.3.4 Simple Recurrent Nets with Binary Neurons

Networks with Uniform Synapses

We now turn to a simple toy model to show how equilibrium statistical mechanics is used for solving neural network models, and to illustrate similarities and differences between the different dynamics types [Coo01, SC00, SC01, CKS05]. We choose uniform infinite-range synapses and zero external fields, and calculate the free energy for the binary systems (3.211,3.212), parallel and sequential, and with threshold-noise distribution $w(\eta) = \frac{1}{2}[1 - \tanh^2(\eta)]$:

$$J_{ij} = J_{ji} = J/N, \quad (i \neq j), \quad J_{ii} = \theta_i = 0, \quad (\text{for all } i).$$

The free energy is an extensive object, $\lim_{N \rightarrow \infty} F/N$ is finite. For the models (3.211,3.212) we now get:

Binary & Sequential Dynamics:

$$\lim_{N \rightarrow \infty} F/N = - \lim_{N \rightarrow \infty} (\beta N)^{-1} \log \sum_{\boldsymbol{\sigma}} e^{\beta N [\frac{1}{2} J m^2(\boldsymbol{\sigma})]},$$

Binary & Parallel Dynamics:

$$\lim_{N \rightarrow \infty} \tilde{F}/N = - \lim_{N \rightarrow \infty} (\beta N)^{-1} \log \sum_{\boldsymbol{\sigma}} e^{N [\log 2 \cosh[\beta J m(\boldsymbol{\sigma})]]},$$

with the average activity $m(\boldsymbol{\sigma}) = \frac{1}{N} \sum_k \sigma_k$. We have to count the number of states $\boldsymbol{\sigma}$ with a prescribed average activity $m = 2n/N - 1$ (n is the number of neurons i with $\sigma_i = 1$), in expressions of the form

$$\begin{aligned} \frac{1}{N} \log \sum_{\boldsymbol{\sigma}} e^{NU[m(\boldsymbol{\sigma})]} &= \frac{1}{N} \log \sum_{n=0}^N \binom{N}{n} e^{NU[2n/N-1]} \\ &= \frac{1}{N} \log \int_{-1}^1 dm e^{N[\log 2 - c^*(m) + U[m]]}, \\ \lim_{N \rightarrow \infty} \frac{1}{N} \log \sum_{\boldsymbol{\sigma}} e^{NU[m(\boldsymbol{\sigma})]} &= \log 2 + \max_{m \in [-1,1]} \{U[m] - c^*(m)\}, \end{aligned}$$

with the *entropic function*

$$c^*(m) = \frac{1}{2}(1+m) \log(1+m) + \frac{1}{2}(1-m) \log(1-m).$$

In order to get there we used Stirling's formula to get the leading term of the factorials (only terms which are exponential in N survive the limit $N \rightarrow \infty$),

we converted (for $N \rightarrow \infty$) the summation over n into an integration over $m = 2n/N - 1 \in [-1, 1]$, and we carried out the integral over m via *saddle-point integration* (see e.g., [Per92]). This leads to a saddle-point problem whose solution gives the free energies [Coo01, SC00, SC01, CKS05]:

$$\lim_{N \rightarrow \infty} F/N = \min_{m \in [-1, 1]} f_{\text{seq}}(m), \quad \beta f_{\text{seq}}(m) = c^*(m) - \log 2 - \frac{1}{2} \beta J m^2. \quad (3.245)$$

$$\lim_{N \rightarrow \infty} \tilde{F}/N = \min_{m \in [-1, 1]} f_{\text{par}}(m), \quad \beta f_{\text{par}}(m) = c^*(m) - 2 \log 2 - \log \cosh[\beta J m]. \quad (3.246)$$

The equations from which to solve the minima are easily got by differentiation, using $\frac{d}{dm} c^*(m) = \tanh^{-1}(m)$. For sequential dynamics we find

$$\text{Binary \& Sequential Dynamics:} \quad m = \tanh[\beta J m], \quad (3.247)$$

which is the so-called *Curie-Weiss law*. For parallel dynamics we find

$$m = \tanh[\beta J \tanh[\beta J m]].$$

One finds that the solutions of the latter equation again obey a Curie-Weiss law. The definition $\hat{m} = \tanh[\beta |J| m]$ transforms it into the coupled equations $m = \tanh[\beta |J| \hat{m}]$ and $\hat{m} = \tanh[\beta |J| m]$, from which we derive

$$0 \leq [m - \hat{m}]^2 = [m - \hat{m}] [\tanh[\beta |J| \hat{m}] - \tanh[\beta |J| m]] \leq 0.$$

Since $\tanh[\beta |J| m]$ is a monotonically increasing function of m , this implies $\hat{m} = m$, so

$$\text{Binary \& Parallel Dynamics:} \quad m = \tanh[\beta |J| m]. \quad (3.248)$$

Our study of the toy models has thus been reduced to analyzing the nonlinear equations (3.247) and (3.248). If $J \geq 0$ (excitation) the two types of dynamics lead to the same behavior. At high noise levels, $T > J$, both minimization problems are solved by $m = 0$, describing a disorganized (paramagnetic) state. This can be seen upon writing the right-hand side of (3.247) in integral form [Coo01, SC00, SC01, CKS05]:

$$m^2 = m \tanh[\beta J m] = \beta J m^2 \int_0^1 dz [1 - \tanh^2[\beta J m z]] \leq \beta J m^2.$$

So $m^2[1 - \beta J] \leq 0$, which gives $m = 0$ as soon as $\beta J < 1$. A phase transition occurs at $T = J$ (a bifurcation of non-trivial solutions of (3.247)), and for $T < J$ the equations for m are solved by the two non-zero solutions of (3.247), describing a state where either all neurons tend to be firing ($m > 0$) or where they tend to be quiet ($m < 0$). This becomes clear when we expand (3.247) for small m : $m = \beta J m + \mathcal{O}(m^3)$, so precisely at $\beta J = 1$ one finds a de-stabilization

of the trivial solution $m = 0$, together with the creation of (two) stable non-trivial ones. Furthermore, using the identity $c^*(\tanh x) = x \tanh x - \log \cosh x$, we get from (3.245, 3.246) the relation $\lim_{N \rightarrow \infty} \tilde{F}/N = 2 \lim_{N \rightarrow \infty} F/N$. For $J < 0$ (inhibition), however, the two types of dynamics give quite different results. For sequential dynamics the relevant minimum is located at $m = 0$ (the paramagnetic state). For parallel dynamics, the minimization problem is invariant under $J \rightarrow -J$, so the behavior is again of the Curie-Weiss type, with a paramagnetic state for $T > |J|$, a phase transition at $T = |J|$, and order for $T < |J|$. This difference between the two types of dynamics for $J < 0$ is explained by studying dynamics. For the present (toy) model in the limit $N \rightarrow \infty$ the average activity evolves in time according to the deterministic laws [Coo01, SC00, SC01, CKS05]

$$\dot{m} = \tanh[\beta J m] - m, \quad m(t+1) = \tanh[\beta J m(t)],$$

for sequential and parallel dynamics, respectively. For $J < 0$ the sequential system always decays towards the trivial state $m = 0$, whereas for sufficiently large β the parallel system enters the stable limit-cycle $m(t) = M_\beta(-1)^t$, where M_β is the non-zero solution of (3.248). The concepts of ‘distance’ and ‘local minima’ are quite different for the two dynamics types; in contrast to the sequential case, parallel dynamics allows the system to make the transition $m \rightarrow -m$ in equilibrium.

Phenomenology of Hopfield Models

Recall that the Hopfield model [Hop82] represents a network of binary neurons of the type (3.211, 3.212), with threshold noise $w(\eta) = \frac{1}{2}[1 - \tanh^2(\eta)]$, and with a specific recipe for the synapses J_{ij} aimed at storing patterns, motivated by suggestions made in the late nineteen-forties [Heb49]. The original model was in fact defined more narrowly, as the zero noise limit of the system (3.212), but the term has since then been accepted to cover a larger network class. Let us first consider the simplest case and try to store a single pattern $\xi \in \{-1, 1\}^N$ in noise-less infinite-range binary networks. Appealing candidates for interactions and thresholds would be $J_{ij} = \xi_i \xi_j$ and $\theta_i = 0$ (for sequential dynamics we put $J_{ii} = 0$ for all i). With this choice the Lyapunov function (3.227) becomes:

$$L_{\text{seq}}[\sigma] = \frac{1}{2}N - \frac{1}{2}[\xi_i \sigma_i]^2.$$

This system indeed reconstructs dynamically the original pattern ξ from an input vector $\sigma(0)$, at least for sequential dynamics. However, *en passant* we have created an additional attractor: the state $-\xi$. This property is shared by all binary models in which the external fields are zero, where the Hamiltonians $H(\sigma)$ (3.239) and $\tilde{H}(\sigma)$ (3.241) are invariant under an overall sign change $\sigma \rightarrow -\sigma$. A second feature common to several (but not all) attractor neural networks is that *each* initial state will lead to pattern reconstruction, even nonsensical (random) ones.

The Hopfield model is got by generalizing the previous simple one-pattern recipe to the case of an arbitrary number p of binary patterns $\xi^\mu = (\xi_1^\mu, \dots, \xi_N^\mu) \in \{-1, 1\}^N$ [Coo01, SC00, SC01, CKS05]:

$$J_{ij} = \frac{1}{N} \xi_i^\mu \xi_j^\mu, \quad \theta_i = 0 \quad (\text{for all } i; \mu = 1, \dots, p), \quad (3.249)$$

(sequential dynamics : $J_{ii} \rightarrow 0$, for all i).

The prefactor N^{-1} has been inserted to ensure that the limit $N \rightarrow \infty$ will exist in future expressions. The process of interest is that where, triggered by correlation between the initial state and a stored pattern ξ^λ , the state vector σ evolves towards ξ^λ . If this happens, pattern ξ^λ is said to be recalled. The similarity between a state vector and the stored patterns is measured by so-called *Hopfield overlaps*

$$m_\mu(\sigma) = \frac{1}{N} \xi_i^\mu \sigma_i. \quad (3.250)$$

The Hopfield model represents as an associative memory, in which the recall process is described in terms of overlaps.

Analysis of Hopfield Models Away From Saturation

A binary Hopfield network with parameters given by (3.249) obeys detailed balance, and the Hamiltonian $H(\sigma)$ (3.239) (corresponding to sequential dynamics) and the pseudo-Hamiltonian $\tilde{H}(\sigma)$ (3.241) (corresponding to parallel dynamics) become [Coo01, SC00, SC01, CKS05]

$$H(\sigma) = -\frac{1}{2} N \sum_{\mu=1}^p m_\mu^2(\sigma) + \frac{1}{2} p, \quad (3.251)$$

$$\tilde{H}(\sigma) = -\frac{1}{\beta} \sum_i \log 2 \cosh[\beta \xi_i^\mu m_\mu(\sigma)],$$

with the overlaps (3.250). Solving the statics implies calculating the free energies F and $\tilde{F}$:

$$F = -\frac{1}{\beta} \log \sum_{\sigma} e^{-\beta H(\sigma)}, \quad \tilde{F} = -\frac{1}{\beta} \log \sum_{\sigma} e^{-\beta \tilde{H}(\sigma)}.$$

Upon introducing the short-hand notation $\mathbf{m} = (m_1, \dots, m_p)$ and $\xi_i = (\xi_i^1, \dots, \xi_i^p)$, both free energies can be expressed in terms of the density of states $\mathcal{D}(\mathbf{m}) = 2^{-N} \sum_{\sigma} \delta[\mathbf{m} - \mathbf{m}(\sigma)]$:

$$F/N = -\frac{1}{\beta} \log 2 - \frac{1}{\beta N} \log \int d\mathbf{m} \mathcal{D}(\mathbf{m}) e^{\frac{1}{2} \beta N \mathbf{m}^2 + \frac{p}{2N}}, \quad (3.252)$$

$$\tilde{F}/N = -\frac{1}{\beta} \log 2 - \frac{1}{\beta N} \log \int d\mathbf{m} \mathcal{D}(\mathbf{m}) e^{\sum_{i=1}^N \log 2 \cosh[\beta \boldsymbol{\xi}_i \cdot \mathbf{m}]}, \quad (3.253)$$

using $\int d\mathbf{m} \delta[\mathbf{m} - \mathbf{m}(\boldsymbol{\sigma})] = 1$. In order to proceed, we need to specify how the number of patterns p scales with the system size N . In this section we will follow [AGS85a] (equilibrium analysis following sequential dynamics) and [FK88] (equilibrium analysis following parallel dynamics), and assume p to be finite. One can now easily calculate the leading contribution to the density of states, using the integral representation of the δ -function and keeping in mind that according to (3.252, 3.253) only terms exponential in N will retain statistical relevance for $N \rightarrow \infty$:

$$\begin{aligned} \lim_{N \rightarrow \infty} \frac{1}{N} \log \mathcal{D}(\mathbf{m}) &= \lim_{N \rightarrow \infty} \frac{1}{N} \log \int d\mathbf{x} e^{iN\mathbf{x} \cdot \mathbf{m}} \langle e^{-i\sigma_i \boldsymbol{\xi}_i \cdot \mathbf{x}} \rangle_{\boldsymbol{\sigma}} \\ &= \lim_{N \rightarrow \infty} \frac{1}{N} \log \int d\mathbf{x} e^{N[i\mathbf{x} \cdot \mathbf{m} + \langle \log \cos[\boldsymbol{\xi} \cdot \mathbf{x}] \rangle_{\boldsymbol{\xi}}]}, \end{aligned}$$

with the abbreviation $\langle \Phi(\boldsymbol{\xi}) \rangle_{\boldsymbol{\xi}} = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{i=1}^N \Phi(\boldsymbol{\xi}_i)$. The leading contribution to both free energies can be expressed as a finite-dimensional integral, for large N dominated by that saddle-point (extremum) for which the extensive exponent is real and maximal [Coo01, SC00, SC01, CKS05]:

$$\begin{aligned} \lim_{N \rightarrow \infty} F/N &= -\frac{1}{\beta N} \log \int d\mathbf{m} d\mathbf{x} e^{-N\beta f(\mathbf{m}, \mathbf{x})} = \text{extr}_{\mathbf{x}, \mathbf{m}} f(\mathbf{m}, \mathbf{x}), \\ \lim_{N \rightarrow \infty} \tilde{F}/N &= -\frac{1}{\beta N} \log \int d\mathbf{m} d\mathbf{x} e^{-N\beta \tilde{f}(\mathbf{m}, \mathbf{x})} = \text{extr}_{\mathbf{x}, \mathbf{m}} \tilde{f}(\mathbf{m}, \mathbf{x}), \quad \text{with} \end{aligned}$$

$$f(\mathbf{m}, \mathbf{x}) = -\frac{1}{2} \mathbf{m}^2 - i\mathbf{x} \cdot \mathbf{m} - \beta^{-1} \langle \log 2 \cos[\beta \boldsymbol{\xi} \cdot \mathbf{x}] \rangle_{\boldsymbol{\xi}},$$

$$\tilde{f}(\mathbf{m}, \mathbf{x}) = -\beta^{-1} \langle \log 2 \cosh[\beta \boldsymbol{\xi} \cdot \mathbf{m}] \rangle_{\boldsymbol{\xi}} - i\mathbf{x} \cdot \mathbf{m} - \beta^{-1} \langle \log 2 \cos[\beta \boldsymbol{\xi} \cdot \mathbf{x}] \rangle_{\boldsymbol{\xi}}.$$

The saddle-point equations for f and $\tilde{f}$ are given by:

$$\begin{aligned} f : \mathbf{x} &= i\mathbf{m}, & i\mathbf{m} &= \langle \boldsymbol{\xi} \tan[\beta \boldsymbol{\xi} \cdot \mathbf{x}] \rangle_{\boldsymbol{\xi}}, \\ \tilde{f} : \mathbf{x} &= i\langle \boldsymbol{\xi} \tanh[\beta \boldsymbol{\xi} \cdot \mathbf{m}] \rangle_{\boldsymbol{\xi}}, & i\mathbf{m} &= \langle \boldsymbol{\xi} \tan[\beta \boldsymbol{\xi} \cdot \mathbf{x}] \rangle_{\boldsymbol{\xi}}. \end{aligned}$$

In saddle-points $\mathbf{x}$ turns out to be purely imaginary. However, after a shift of the integration contours, putting $\mathbf{x} = i\mathbf{x}^*(\mathbf{m}) + \mathbf{y}$ (where $i\mathbf{x}^*(\mathbf{m})$ is the imaginary saddle-point, and where $\mathbf{y} \in \mathbb{R}^p$) we can eliminate $\mathbf{x}$ in favor of $\mathbf{y} \in \mathbb{R}^p$ which does have a real saddle-point, by construction. (Our functions to be integrated have no poles, but strictly speaking we still have to verify that the integration segments linking the original integration regime to the shifted one will not contribute to the integrals. This is generally a tedious and distracting task, which is often skipped. For simple models, however (e.g., networks with uniform synapses), the verification can be carried out properly, and all is found to be safe.) We then get

Sequential Dynamics: $\mathbf{m} = \langle \boldsymbol{\xi} \tanh[\beta \boldsymbol{\xi} \cdot \mathbf{m}] \rangle_{\boldsymbol{\xi}},$

Parallel Dynamics: $\mathbf{m} = \langle \boldsymbol{\xi} \tanh[\beta \boldsymbol{\xi} \cdot [\langle \boldsymbol{\xi}' \tanh[\beta \boldsymbol{\xi}' \cdot \mathbf{m}] \rangle_{\boldsymbol{\xi}'}]] \rangle_{\boldsymbol{\xi}},$

(compare to e.g., (3.247, 3.248)). The solutions of the above two equations will in general be identical. To see this, let us denote $\hat{\mathbf{m}} = \langle \boldsymbol{\xi} \tanh[\beta \boldsymbol{\xi} \cdot \mathbf{m}] \rangle_{\boldsymbol{\xi}}$, with which the saddle point equation for $\tilde{f}$ decouples into:

$$\begin{aligned} \mathbf{m} &= \langle \boldsymbol{\xi} \tanh[\beta \boldsymbol{\xi} \cdot \hat{\mathbf{m}}] \rangle_{\boldsymbol{\xi}}, & \hat{\mathbf{m}} &= \langle \boldsymbol{\xi} \tanh[\beta \boldsymbol{\xi} \cdot \mathbf{m}] \rangle_{\boldsymbol{\xi}}, & \text{so} \\ [\mathbf{m} - \hat{\mathbf{m}}]^2 &= \langle [(\boldsymbol{\xi} \cdot \mathbf{m}) - (\boldsymbol{\xi} \cdot \hat{\mathbf{m}})] [\tanh(\beta \boldsymbol{\xi} \cdot \hat{\mathbf{m}}) - \tanh(\beta \boldsymbol{\xi} \cdot \mathbf{m})] \rangle_{\boldsymbol{\xi}}. \end{aligned}$$

Since $\tanh$ is a monotonically-increasing function, we must have $[\mathbf{m} - \hat{\mathbf{m}}] \cdot \boldsymbol{\xi} = 0$ for each $\boldsymbol{\xi}$ that contributes to the averages $\langle \dots \rangle_{\boldsymbol{\xi}}$. For all choices of patterns where the covariance matrix $C_{\mu\nu} = \langle \xi_{\mu} \xi_{\nu} \rangle_{\boldsymbol{\xi}}$ is positive definite, we thus get $\mathbf{m} = \hat{\mathbf{m}}$. The final result is: for both types of dynamics (sequential and parallel) the overlap order parameters in equilibrium are given by the solution $\mathbf{m}^*$ of

$$\mathbf{m} = \langle \boldsymbol{\xi} \tanh[\beta \boldsymbol{\xi} \cdot \mathbf{m}] \rangle_{\boldsymbol{\xi}}, \quad \text{which minimizes} \quad (3.254)$$

$$f(\mathbf{m}) = \frac{1}{2} \mathbf{m}^2 - \frac{1}{\beta} \langle \log 2 \cosh[\beta \boldsymbol{\xi} \cdot \mathbf{m}] \rangle_{\boldsymbol{\xi}}. \quad (3.255)$$

The free energies of the ergodic components are $\lim_{N \rightarrow \infty} F/N = f(\mathbf{m}^*)$ and $\lim_{N \rightarrow \infty} \tilde{F}/N = 2f(\mathbf{m}^*)$. Adding partition terms of the form $H \rightarrow H + \lambda g[\mathbf{m}(\boldsymbol{\sigma})]$ to the Hamiltonians allows us identify $\langle g[\mathbf{m}(\boldsymbol{\sigma})] \rangle_{\text{eq}} = \lim_{\lambda \rightarrow 0} \partial F / \partial \lambda = g[\mathbf{m}^*]$. Thus, in equilibrium the fluctuations in the overlap order parameters $\mathbf{m}(\boldsymbol{\sigma})$ (3.250) vanish for $N \rightarrow \infty$. Their deterministic values are simply given by $\mathbf{m}^*$. Note that in the case of sequential dynamics we could also have used linearization with Gaussian integrals (as used previously for coupled oscillators with uniform synapses) to arrive at this solution, with p auxiliary integrations, but that for parallel dynamics this would not have been possible.

Now, in analysis of *order parameter equations*, we will restrict our further discussion to the case of randomly drawn patterns, so [Coo01, SC00, SC01, CKS05]

$$\langle \Phi(\boldsymbol{\xi}) \rangle_{\boldsymbol{\xi}} = 2^{-p} \sum_{\boldsymbol{\xi} \in \{-1, 1\}^p} \Phi(\boldsymbol{\xi}), \quad \langle \xi_{\mu} \rangle_{\boldsymbol{\xi}} = 0, \quad \langle \xi_{\mu} \xi_{\nu} \rangle_{\boldsymbol{\xi}} = \delta_{\mu\nu}.$$

We first establish an upper bound for the temperature for where non-trivial solutions $\mathbf{m}^*$ could exist, by writing (3.254) in integral form:

$$m_{\mu} = \beta \langle \xi_{\mu}(\boldsymbol{\xi} \cdot \mathbf{m}) \int_0^1 d\lambda [1 - \tanh^2[\beta \lambda \boldsymbol{\xi} \cdot \mathbf{m}]] \rangle_{\boldsymbol{\xi}},$$

from which we deduce

$$\begin{aligned} 0 &= \mathbf{m}^2 - \beta \langle (\boldsymbol{\xi} \cdot \mathbf{m})^2 \int_0^1 d\lambda [1 - \tanh^2[\beta \lambda \boldsymbol{\xi} \cdot \mathbf{m}]] \rangle_{\boldsymbol{\xi}} \\ &\geq \mathbf{m}^2 - \beta \langle (\boldsymbol{\xi} \cdot \mathbf{m})^2 \rangle_{\boldsymbol{\xi}} = \mathbf{m}^2(1 - \beta), \end{aligned}$$

For $T > 1$ the only solution of (3.254) is the paramagnetic state $\mathbf{m} = 0$, which gives for the free energy per neuron $-T \log 2$ and $-2T \log 2$ (for sequential and parallel dynamics, respectively). At $T = 1$ a phase transition occurs, which follows from expanding (3.254) for small $|\mathbf{m}|$ in powers of $\tau = \beta - 1$:

$$m_\mu = (1 + \tau)m_\mu - \frac{1}{3}m_\nu m_\rho m_\lambda \langle \xi_\mu \xi_\nu \xi_\rho \xi_\lambda \rangle_\xi \\ + \mathcal{O}(\mathbf{m}^5, \tau \mathbf{m}^3) = m_\mu [1 + \tau - \mathbf{m}^2 + \frac{2}{3}m_\mu^2] + \mathcal{O}(\mathbf{m}^5, \tau \mathbf{m}^3).$$

The new saddle-point scales as $m_\mu = \tilde{m}_\mu \tau^{1/2} + \mathcal{O}(\tau^{3/2})$, with for each μ : $\tilde{m}_\mu = 0$ or $0 = 1 - \tilde{\mathbf{m}}^2 + \frac{2}{3}\tilde{m}_\mu^2$.

The solutions are of the form $\tilde{m}_\mu \in \{-\tilde{m}, 0, \tilde{m}\}$. If we denote with n the number of non-zero components in the vector $\tilde{\mathbf{m}}$, we derive from the above identities: $\tilde{m}_\mu = 0$ or $\tilde{m}_\mu = \pm\sqrt{3}/\sqrt{3n-2}$. These saddle-points are called *mixture states*, since they correspond to microscopic configurations correlated equally with a finite number n of the stored patterns (or their negatives). Without loss of generality we can always perform gauge transformations on the set of stored patterns (permutations and reflections), such that the mixture states acquire the form [Coo01, SC00, SC01, CKS05]

$$\mathbf{m} = m_n \left(\overbrace{1, \dots, 1}^{n \text{ times}}, \overbrace{0, \dots, 0}^{p-n \text{ times}} \right), \quad (3.256) \\ m_n = \left[\frac{3}{3n-2} \right]^{\frac{1}{2}} (\beta - 1)^{1/2} + \dots$$

These states are in fact saddle-points of the surface $f(\mathbf{m})$ (3.255) for any finite temperature, as can be verified by substituting (3.256) as an *ansatz* into (3.254):

$$\mu \leq n : \quad m_n = \langle \xi_\mu \tanh[\beta m_n \sum_{\nu \leq n} \xi_\nu] \rangle_\xi, \\ \mu > n : \quad 0 = \langle \xi_\mu \tanh[\beta m_n \sum_{\nu \leq n} \xi_\nu] \rangle_\xi.$$

The second equation is automatically satisfied since the average factorizes. The first equation leads to a condition determining the amplitude m_n of the mixture states:

$$m_n = \langle \left[\frac{1}{n} \sum_{\mu \leq n} \xi_\mu \right] \tanh[\beta m_n \sum_{\nu \leq n} \xi_\nu] \rangle_\xi. \quad (3.257)$$

The corresponding values of $f(\mathbf{m})$, to be denoted by f_n , are

$$f_n = \frac{1}{2} n m_n^2 - \frac{1}{\beta} \langle \log 2 \cosh[\beta m_n \sum_{\nu \leq n} \xi_\nu] \rangle_\xi. \quad (3.258)$$

The relevant question at this stage is whether or not these saddle-points correspond to local minima of the surface $f(\mathbf{m})$ (3.255). The second derivative of $f(\mathbf{m})$ is given by

$$\frac{\partial^2 f(\mathbf{m})}{\partial m_\mu \partial m_\nu} = \delta_{\mu\nu} - \beta \langle \xi_\mu \xi_\nu [1 - \tanh^2 [\beta \boldsymbol{\xi} \cdot \mathbf{m}]] \rangle_{\boldsymbol{\xi}}, \quad (3.259)$$

where a local minimum corresponds to a positive definite second derivative. In the trivial saddle-point $\mathbf{m} = 0$ this gives simply $\delta_{\mu\nu}(1 - \beta)$, so at $T = 1$ this state destabilizes. In a mixture state of the type (3.256) the second derivative becomes:

$$D_{\mu\nu}^{(n)} = \delta_{\mu\nu} - \beta \langle \xi_\mu \xi_\nu [1 - \tanh^2 [\beta m_n \sum_{\rho \leq n} \xi_\rho]] \rangle_{\boldsymbol{\xi}}.$$

Due to the symmetries in the problem the spectrum of the matrix $D^{(n)}$ can be calculated. One finds the following eigen-spaces, with

$$Q = \langle \tanh^2 [\beta m_n \sum_{\rho \leq n} \xi_\rho] \rangle_{\boldsymbol{\xi}} \quad \text{and} \quad R = \langle \xi_1 \xi_2 \tanh^2 [\beta m_n \sum_{\rho \leq n} \xi_\rho] \rangle_{\boldsymbol{\xi}},$$

Eigenspace :

Eigenvalue :

$$I : \quad \mathbf{x} = (0, \dots, 0, x_{n+1}, \dots, x_p),$$

$$1 - \beta[1 - Q],$$

$$II : \quad \mathbf{x} = (1, \dots, 1, 0, \dots, 0),$$

$$1 - \beta[1 - Q + (1 - n)R],$$

$$III : \quad \mathbf{x} = (x_1, \dots, x_n, 0, \dots, 0), \quad \sum_{\mu} x_\mu = 0, \quad 1 - \beta[1 - Q + R].$$

The eigen-space *III* and the quantity R only come into play for $n > 1$. To find the smallest eigenvalue we need to know the sign of R . With the abbreviation $M_{\boldsymbol{\xi}} = \sum_{\rho \leq n} \xi_\rho$ we find [Coo01, SC00, SC01, CKS05]:

$$\begin{aligned} n(n-1)R &= \langle M_{\boldsymbol{\xi}}^2 \tanh^2 [\beta m_n M_{\boldsymbol{\xi}}] \rangle_{\boldsymbol{\xi}} - n \langle \tanh^2 [\beta m_n M_{\boldsymbol{\xi}}] \rangle_{\boldsymbol{\xi}} \\ &= \langle [M_{\boldsymbol{\xi}}^2 - \langle M_{\boldsymbol{\xi}'}^2 \rangle_{\boldsymbol{\xi}'}] \tanh^2 [\beta m_n M_{\boldsymbol{\xi}}] \rangle_{\boldsymbol{\xi}} \\ &= \langle [M_{\boldsymbol{\xi}}^2 - \langle M_{\boldsymbol{\xi}'}^2 \rangle_{\boldsymbol{\xi}'}] \left\{ \tanh^2 [\beta m_n \sqrt{M_{\boldsymbol{\xi}}^2}] - \tanh^2 [\beta m_n \sqrt{\langle M_{\boldsymbol{\xi}'}^2 \rangle_{\boldsymbol{\xi}'}}] \right\} \rangle_{\boldsymbol{\xi}} \\ &\geq 0. \end{aligned}$$

We may now identify the conditions for an n -mixture state to be a local minimum of $f(\mathbf{m})$. For $n = 1$ the relevant eigenvalue is I , now the quantity Q simplifies considerably. For $n > 1$ the relevant eigenvalue is *III*, here we can combine Q and R into one single average:

$$n = 1 : 1 - \beta[1 - \tanh^2 [\beta m_1]] > 0$$

$$n = 2 : 1 - \beta > 0$$

$$n \geq 3 : 1 - \beta[1 - \langle \tanh^2 [\beta m_n \sum_{\rho=3}^n \xi_\rho] \rangle_{\boldsymbol{\xi}}] > 0$$

The $n = 1$ states, correlated with one pattern only, are the desired solutions. They are stable for all $T < 1$, since partial differentiation with respect to β of the $n = 1$ amplitude equation (3.257) gives

$$\begin{aligned} m_1 &= \tanh[\beta m_1] \rightarrow 1 - \beta[1 - \tanh^2 [\beta m_1]] \\ &= m_1[1 - \tanh^2 [\beta m_1]](\partial m_1 / \partial \beta)^{-1}, \end{aligned}$$

so that clearly $\text{sgn}[m_1] = \text{sgn}[\partial m_1 / \partial \beta]$. The $n = 2$ mixtures are always unstable. For $n \geq 3$ we have to solve the amplitude equations (3.257) numerically to evaluate their stability. It turns out that only for odd n will there be a critical temperature below which the n -mixture states are local minima of $f(\mathbf{m})$.

We have now solved the model in equilibrium for finite p and $N \rightarrow \infty$. For non-random patterns one simply has to study the bifurcation properties of equation (3.254) for the new pattern statistics at hand; this is only qualitatively different from the random pattern analysis explained above. The occurrence of multiple saddle-points corresponding to local minima of the free energy signals ergodicity breaking. Although among these only the *global minimum* will correspond to the thermodynamic equilibrium state, the non-global minima correspond to true ergodic components, i.e., on finite time-scales they will be just as relevant as the global minimum.

3.3.5 Simple Recurrent Nets of Coupled Oscillators

Coupled Oscillators with Uniform Synapses

Models with continuous variables involve integration over states, rather than summation. For a coupled oscillator network (3.223) with uniform synapses $J_{ij} = J/N$ and zero frequencies $\omega_i = 0$ (which is a simple version of the model in [Kur84]) we get for the free energy per oscillator [Coo01, SC00, SC01, CKS05]:

$$\lim_{N \rightarrow \infty} F/N = - \lim_{N \rightarrow \infty} \frac{1}{\beta N} \log \int_{-\pi}^{\pi} \cdots \int_{-\pi}^{\pi} d\xi \\ \times e^{(\beta J/2N) [\sum_i \cos(\xi_i)]^2 + [\sum_i \sin(\xi_i)]^2}.$$

We would now have to ‘count’ microscopic states with prescribed average cosines and sines. A faster route exploits auxiliary Gaussian integrals, via the identity

$$e^{\frac{1}{2}y^2} = \int Dz e^{yz}, \quad (3.260)$$

with the short-hand $Dx = (2\pi)^{-\frac{1}{2}} e^{-\frac{1}{2}x^2} dx$ (this alternative would also have been open to us in the binary case; my aim in this section is to explain both methods):

$$\begin{aligned}
\lim_{N \rightarrow \infty} F/N &= - \lim_{N \rightarrow \infty} \frac{1}{\beta N} \log \int_{-\pi}^{\pi} \cdots \int_{-\pi}^{\pi} d\xi \int Dx Dy \\
&\quad \times e^{\sqrt{\beta J/N} [x \sum_i \cos(\xi_i) + y \sum_i \sin(\xi_i)]} \\
&= - \lim_{N \rightarrow \infty} \frac{1}{\beta N} \log \int Dx Dy \left[\int_{-\pi}^{\pi} d\xi e^{\cos(\xi) \sqrt{\beta J(x^2 + y^2)/N}} \right]^N \\
&= - \lim_{N \rightarrow \infty} \frac{1}{\beta N} \log \int_0^\infty dq q e^{-\frac{1}{2} N \beta |J| q^2} \\
&\quad \times \left[\int_{-\pi}^{\pi} d\xi e^{\beta |J| q \cos(\xi) \sqrt{\text{rmsgn}(J)}} \right]^N,
\end{aligned}$$

where we have transformed to polar coordinates, $(x, y) = q\sqrt{\beta|J|N}(\cos \theta, \sin \theta)$, and where we have already eliminated (constant) terms which will not survive the limit $N \rightarrow \infty$. Thus, saddle-point integration gives us, quite similar to the previous cases (3.245, 3.246):

$$\begin{aligned}
\lim_{N \rightarrow \infty} F/N &= \min_{q \geq 0} f(q), & J > 0 : \beta f(q) &= \frac{1}{2} \beta |J| q^2 - \log[2\pi I_0(\beta |J| q)] \\
& & J < 0 : \beta f(q) &= \frac{1}{2} \beta |J| q^2 - \log[2\pi I_0(i\beta |J| q)]',
\end{aligned} \tag{3.261}$$

in which the $I_n(z)$ are the Bessel functions (see e.g., [AS72]). The equations from which to solve the minima are got by differentiation, using $\frac{d}{dz} I_0(z) = I_1(z)$: in which the $I_n(z)$ are the *Bessel functions* (see e.g., [AS72]). The equations from which to solve the minima are got by differentiation, using $\frac{d}{dz} I_0(z) = I_1(z)$:

$$J > 0 : \quad q = \frac{I_1(\beta |J| q)}{I_0(\beta |J| q)}, \quad J < 0 : \quad q = i \frac{I_1(i\beta |J| q)}{I_0(i\beta |J| q)}. \tag{3.262}$$

Again, in both cases the problem has been reduced to studying a single non-linear equation. The physical meaning of the solution follows from the identity $-2\partial F/\partial J = \langle N^{-1} \sum_{i \neq j} \cos(\xi_i - \xi_j) \rangle$:

$$\lim_{N \rightarrow \infty} \langle [\frac{1}{N} \sum_i \cos(\xi_i)]^2 \rangle + \lim_{N \rightarrow \infty} \langle [\frac{1}{N} \sum_i \sin(\xi_i)]^2 \rangle = \text{sgn}(J) q^2.$$

From this equation it also follows that $q \leq 1$. Note: since $\partial f(q)/\partial q = 0$ at the minimum, one only needs to consider the explicit derivative of $f(q)$ with respect to J . If the synapses induce anti-synchronization, $J < 0$, the only solution of (3.262) (and the minimum in (3.261)) is the trivial state $q = 0$. This also follows immediately from the equation which gave the physical meaning of q . For synchronizing forces, $J > 0$, on the other hand, we again find the trivial solution at high noise levels, but a globally synchronized state with $q > 0$ at low noise levels. Here a phase transition occurs at $T = \frac{1}{2}J$ (a bifurcation of non-trivial solutions of (3.262)), and for $T < \frac{1}{2}J$ the minimum of (3.261) is

found at two non-zero values for q . The critical noise level is again found upon expanding the saddle-point equation, using

$$I_0(z) = 1 + \mathcal{O}(z^2) \quad \text{and} \quad I_1(z) = \frac{1}{2}z + \mathcal{O}(z^3) : q = \frac{1}{2}\beta Jq + \mathcal{O}(q^3).$$

Precisely at $\beta J = 2$ one finds a de-stabilization of the trivial solution $q = 0$, together with the creation of (two) stable non-trivial ones. Note that, in view of (3.261), we are only interested in non-negative values of q . One can prove, using the properties of the Bessel functions, that there are no other (discontinuous) bifurcations of non-trivial solutions of the saddle-point equation. Note, finally, that the absence of a state with global anti-synchronization for $J < 0$ has the same origin as the absence of an anti-ferromagnetic state for $J < 0$ in the previous models with binary neurons. Due to the long-range nature of the synapses $J_{ij} = J/N$ such states simply cannot exist: whereas any set of oscillators can be in a fully synchronized state, if two oscillators are in anti-synchrony it is already impossible for a third to be simultaneously in anti-synchrony with the first two (since anti-synchrony with one implies synchrony with the other) [Coo01, SC00, SC01, CKS05].

Coupled Oscillator Attractor Networks

Let us now turn to an alternative realization of information storage in a recurrent network based upon the creation of attractors [Coo01, SC00, SC01, CKS05]. We will solve models of coupled neural oscillators of the type (3.223), with zero natural frequencies (since we wish to use equilibrium techniques), in which real-valued patterns are stored as stable configurations of oscillator phases, following [Coo89]. Let us, however, first find out how to store a single pattern $\boldsymbol{\xi} \in [-\pi, \pi]^N$ in a noise-less infinite-range oscillator network. For simplicity we will draw each component ξ_i independently at random from $[-\pi, \pi]$, with uniform probability density. This allows us to use asymptotic properties such as $|N^{-1} \sum_j e^{i\ell\xi_j}| = \mathcal{O}(N^{-\frac{1}{2}})$ for any integer ℓ . A sensible choice for the synapses would be $J_{ij} = \cos[\xi_i - \xi_j]$. To see this we work out the corresponding Lyapunov function (3.230):

$$\begin{aligned} L[\boldsymbol{\xi}] &= -\frac{1}{2N^2} \cos[\xi_i - \xi_j] \cos[\xi_i - \xi_j], \\ L[\boldsymbol{\xi}] &= -\frac{1}{2N^2} \cos^2[\xi_i - \xi_j] = -\frac{1}{4} + \mathcal{O}(N^{-\frac{1}{2}}), \end{aligned}$$

where the factors of N have been inserted to achieve appropriate scaling in the $N \rightarrow \infty$ limit. The function $L[\boldsymbol{\xi}]$, which is obviously bounded from below, must decrease monotonically during the dynamics. To find out whether the state $\boldsymbol{\xi}$ is a stable fixed-point of the dynamics we have to calculate L and derivatives of L at $\boldsymbol{\xi} = \boldsymbol{\xi}$:

$$\begin{aligned}\left.\frac{\partial L}{\partial \xi_i}\right|_{\boldsymbol{\xi}} &= \frac{1}{2N^2} \sum_j \sin[2(\xi_i - \xi_j)], \\ \left.\frac{\partial^2 L}{\partial \xi_i^2}\right|_{\boldsymbol{\xi}} &= \frac{1}{N^2} \sum_j \cos^2[\xi_i - \xi_j], \\ i \neq j: \quad \left.\frac{\partial^2 L}{\partial \xi_i \partial \xi_j}\right|_{\boldsymbol{\xi}} &= -\frac{1}{N^2} \cos^2[\xi_i - \xi_j].\end{aligned}$$

Clearly $\lim_{N \rightarrow \infty} L[\boldsymbol{\xi}] = -\frac{1}{4}$. Putting $\boldsymbol{\xi} = \boldsymbol{\xi} + \Delta\boldsymbol{\xi}$, with $\Delta\xi_i = \mathcal{O}(N^0)$, we find

$$\begin{aligned}L[\boldsymbol{\xi} + \Delta\boldsymbol{\xi}] - L[\boldsymbol{\xi}] &= \Delta\xi_i \left.\frac{\partial L}{\partial \xi_i}\right|_{\boldsymbol{\xi}} \\ &+ \frac{1}{2} \Delta\xi_i \Delta\xi_j \left.\frac{\partial^2 L}{\partial \xi_i \partial \xi_j}\right|_{\boldsymbol{\xi}} + \mathcal{O}(\Delta\boldsymbol{\xi}^3) \\ &= \frac{1}{4N} \sum_i \Delta\xi_i^2 - \frac{1}{2N^2} \Delta\xi_i \Delta\xi_j \cos^2[\xi_i - \xi_j] + \mathcal{O}(N^{-\frac{1}{2}}, \Delta\boldsymbol{\xi}^3) \\ &= \frac{1}{4} \left\{ \frac{1}{N} \sum_i \Delta\xi_i^2 - \left[\frac{1}{N} \sum_i \Delta\xi_i \right]^2 - \left[\frac{1}{N} \Delta\xi_i \cos(2\xi_i) \right]^2 \right. \\ &\quad \left. - \left[\frac{1}{N} \sum_i \Delta\xi_i \sin(2\xi_i) \right]^2 \right\} + \mathcal{O}(N^{-\frac{1}{2}}, \Delta\boldsymbol{\xi}^3).\end{aligned}\tag{3.263}$$

In leading order in N the following three vectors in $\mathbb{R}^N$ are normalized and orthogonal:

$$\begin{aligned}\mathbf{e}_1 &= \frac{1}{\sqrt{N}}(1, 1, \dots, 1), \quad \mathbf{e}_2 = \frac{\sqrt{2}}{\sqrt{N}}(\cos(2\xi_1), \dots, \cos(2\xi_N)), \\ \mathbf{e}_3 &= \frac{\sqrt{2}}{\sqrt{N}}(\sin(2\xi_1), \dots, \sin(2\xi_N)).\end{aligned}$$

We may therefore use

$$\Delta\boldsymbol{\xi}^2 \geq (\Delta\boldsymbol{\xi} \cdot \mathbf{e}_1)^2 + (\Delta\boldsymbol{\xi} \cdot \mathbf{e}_2)^2 + (\Delta\boldsymbol{\xi} \cdot \mathbf{e}_3)^2,$$

insertion of which into (3.263) leads to

$$\begin{aligned}L[\boldsymbol{\xi} + \Delta\boldsymbol{\xi}] - L[\boldsymbol{\xi}] &\geq \left[\frac{1}{2N} \sum_i \Delta\xi_i \cos(2\xi_i) \right]^2 \\ &+ \left[\frac{1}{2N} \sum_i \Delta\xi_i \sin(2\xi_i) \right]^2 + \mathcal{O}(N^{-\frac{1}{2}}, \Delta\boldsymbol{\xi}^3).\end{aligned}$$

Thus for large N the second derivative of L is non-negative at $\boldsymbol{\xi} = \boldsymbol{\xi}$, and the phase pattern $\boldsymbol{\xi}$ has indeed become a fixed-point attractor of the dynamics

of the noise-free coupled oscillator network. The same is found to be true for the states $\boldsymbol{\xi} = \pm \boldsymbol{\xi} + \alpha(1, \dots, 1)$ (for any α).

We next follow the strategy of the Hopfield model and attempt to simply extend the above recipe for the synapses to the case of having a finite number p of phase patterns $\boldsymbol{\xi}^\mu = (\xi_1^\mu, \dots, \xi_N^\mu) \in [-\pi, \pi]^N$, giving

$$J_{ij} = \frac{1}{N} \sum_{\mu=1}^p \cos[\xi_i^\mu - \xi_j^\mu], \quad (3.264)$$

where the factor N , as before, ensures a proper limit $N \rightarrow \infty$ later. In analogy with our solution of the Hopfield model we define the following averages over pattern variables:

$$\langle g[\boldsymbol{\xi}] \rangle_{\boldsymbol{\xi}} = \lim_{N \rightarrow \infty} \sum_i g[\xi_i], \quad \boldsymbol{\xi}_i = (\xi_i^1, \dots, \xi_i^p) \in [-\pi, \pi]^p.$$

We can write the Hamiltonian $H(\boldsymbol{\xi})$ of (3.243) in the form [Coo01, SC00, SC01, CKS05]

$$\begin{aligned} H(\boldsymbol{\xi}) &= -\frac{1}{2N} \sum_{\mu=1}^p \cos[\xi_i^\mu - \xi_j^\mu] \cos[\xi_i - \xi_j] \\ &= -\frac{N}{2} \sum_{\mu=1}^p \{m_{cc}^\mu(\boldsymbol{\xi})^2 + m_{cs}^\mu(\boldsymbol{\xi})^2 + m_{sc}^\mu(\boldsymbol{\xi})^2 + m_{ss}^\mu(\boldsymbol{\xi})^2\}, \end{aligned}$$

in which

$$\begin{aligned} m_{cc}^\mu(\boldsymbol{\xi}) &= \frac{1}{N} \cos(\xi_i^\mu) \cos(\xi_i), \\ m_{cs}^\mu(\boldsymbol{\xi}) &= \frac{1}{N} \cos(\xi_i^\mu) \sin(\xi_i), \end{aligned} \quad (3.265)$$

$$\begin{aligned} m_{sc}^\mu(\boldsymbol{\xi}) &= \frac{1}{N} \sin(\xi_i^\mu) \cos(\xi_i), \\ m_{ss}^\mu(\boldsymbol{\xi}) &= \frac{1}{N} \sin(\xi_i^\mu) \sin(\xi_i). \end{aligned} \quad (3.266)$$

The free energy per oscillator can now be written as

$$\begin{aligned} F/N &= -\frac{1}{\beta N} \log \int \dots \int d\boldsymbol{\xi} e^{-\beta H(\boldsymbol{\xi})} = \\ &= -\frac{1}{\beta N} \log \int \dots \int d\boldsymbol{\xi} e^{\frac{1}{2}\beta N \sum_\mu \sum_{\star\star} m_{\star\star}^\mu(\boldsymbol{\xi})^2}, \end{aligned}$$

with $\star\star \in \{cc, ss, cs, sc\}$. Upon introducing the notation $\mathbf{m}_{\star\star} = (m_{\star\star}^1, \dots, m_{\star\star}^p)$ we can again express the free energy in terms of the density of states $\mathcal{D}(\{\mathbf{m}_{\star\star}\}) = (2\pi)^{-N} \int \dots \int d\boldsymbol{\xi} \prod_{\star\star} \delta[\mathbf{m}_{\star\star} - \mathbf{m}_{\star\star}(\boldsymbol{\sigma})]$:

$$F/N = -\frac{1}{\beta} \log(2\pi) - \frac{1}{\beta N} \log \int \prod_{**} d\mathbf{m}_{**} \mathcal{D}(\{\mathbf{m}_{**}\}) e^{\frac{1}{2}\beta N \sum_{**} \mathbf{m}_{**}^2}. \quad (3.267)$$

Since p is finite, the leading contribution to the density of states (as $N \rightarrow \infty$), which will give us the entropy, can be calculated by writing the δ -functions in integral representation:

$$\begin{aligned} \lim_{N \rightarrow \infty} \frac{1}{N} \log \mathcal{D}(\{\mathbf{m}_{**}\}) &= \lim_{N \rightarrow \infty} \frac{1}{N} \log \int \prod_{**} [d\mathbf{x}_{**} e^{iN\mathbf{x}_{**} \cdot \mathbf{m}_{**}}] \times \\ &\int \cdots \int \frac{d\xi}{(2\pi)^N} \times \\ &e^{-i[x_{cc}^\mu \cos(\xi_i^\mu) \cos(\xi_i) + x_{cs}^\mu \cos(\xi_i^\mu) \sin(\xi_i) + x_{sc}^\mu \sin(\xi_i^\mu) \cos(\xi_i) + x_{ss}^\mu \sin(\xi_i^\mu) \sin(\xi_i)]} \\ &= \text{extr}_{\{\mathbf{x}_{**}\}} \{i \sum_{**} \mathbf{x}_{**} \cdot \mathbf{m}_{**} + \langle \log \int \frac{d\xi}{2\pi} \times \\ &e^{-i[x_{cc}^\mu \cos(\xi_\mu) \cos(\xi) + x_{cs}^\mu \cos(\xi_\mu) \sin(\xi) + x_{sc}^\mu \sin(\xi_\mu) \cos(\xi) + x_{ss}^\mu \sin(\xi_\mu) \sin(\xi)]} \rangle_{\xi}\}. \end{aligned}$$

The relevant extremum is purely imaginary so we put $\mathbf{x}_{**} = i\beta\mathbf{y}_{**}$ (see our previous discussion for the Hopfield model) and, upon inserting the density of states into our original expression for the free energy per oscillator, arrive at

$$\begin{aligned} \lim_{N \rightarrow \infty} F/N &= \text{extr}_{\{\mathbf{m}_{**}, \mathbf{y}_{**}\}} f(\{\mathbf{m}_{**}, \mathbf{y}_{**}\}), \\ f(\{\mathbf{m}_{**}, \mathbf{y}_{**}\}) &= -\frac{1}{\beta} \log(2\pi) - \frac{1}{2} \sum_{**} \mathbf{m}_{**}^2 + \sum_{**} \mathbf{y}_{**} \cdot \mathbf{m}_{**} \\ &- \frac{1}{\beta} \langle \log \int \frac{d\xi}{2\pi} e^{\beta[y_{cc}^\mu \cos(\xi_\mu) \cos(\xi) + y_{cs}^\mu \cos(\xi_\mu) \sin(\xi) + y_{sc}^\mu \sin(\xi_\mu) \cos(\xi) + y_{ss}^\mu \sin(\xi_\mu) \sin(\xi)]} \rangle_{\xi} \end{aligned}$$

Taking derivatives with respect to the order parameters $\mathbf{m}_{**}$ gives us $\mathbf{y}_{**} = \mathbf{m}_{**}$, with which we can eliminate the $\mathbf{y}_{**}$. Derivation with respect to the $\mathbf{m}_{**}$ subsequently gives the saddle-point equations

$$\begin{aligned} m_{cc}^\mu &= \quad (3.268) \\ \langle \cos[\xi_\mu] \frac{\int d\xi \cos[\xi] e^{\beta \cos[\xi] [m_{cc}^\nu \cos[\xi_\nu] + m_{sc}^\nu \sin[\xi_\nu]] + \beta \sin[\xi] [m_{cs}^\nu \cos[\xi_\nu] + m_{ss}^\nu \sin[\xi_\nu]]}}{\int d\xi e^{\beta \cos[\xi] [m_{cc}^\nu \cos[\xi_\nu] + m_{sc}^\nu \sin[\xi_\nu]] + \beta \sin[\xi] [m_{cs}^\nu \cos[\xi_\nu] + m_{ss}^\nu \sin[\xi_\nu]]}} \rangle_{\xi}, \end{aligned}$$

$$\begin{aligned} m_{cs}^\mu &= \quad (3.269) \\ \langle \cos[\xi_\mu] \frac{\int d\xi \sin[\xi] e^{\beta \cos[\xi] [m_{cc}^\nu \cos[\xi_\nu] + m_{sc}^\nu \sin[\xi_\nu]] + \beta \sin[\xi] [m_{cs}^\nu \cos[\xi_\nu] + m_{ss}^\nu \sin[\xi_\nu]]}}{\int d\xi e^{\beta \cos[\xi] [m_{cc}^\nu \cos[\xi_\nu] + m_{sc}^\nu \sin[\xi_\nu]] + \beta \sin[\xi] [m_{cs}^\nu \cos[\xi_\nu] + m_{ss}^\nu \sin[\xi_\nu]]}} \rangle_{\xi}, \end{aligned}$$

$$\begin{aligned} m_{sc}^\mu &= \quad (3.270) \\ \langle \sin[\xi_\mu] \frac{\int d\xi \cos[\xi] e^{\beta \cos[\xi] [m_{cc}^\nu \cos[\xi_\nu] + m_{sc}^\nu \sin[\xi_\nu]] + \beta \sin[\xi] [m_{cs}^\nu \cos[\xi_\nu] + m_{ss}^\nu \sin[\xi_\nu]]}}{\int d\xi e^{\beta \cos[\xi] [m_{cc}^\nu \cos[\xi_\nu] + m_{sc}^\nu \sin[\xi_\nu]] + \beta \sin[\xi] [m_{cs}^\nu \cos[\xi_\nu] + m_{ss}^\nu \sin[\xi_\nu]]}} \rangle_{\xi}, \end{aligned}$$

$$m_{ss}^\mu = \langle \sin[\xi_\mu] \frac{\int d\xi \sin[\xi] e^{\beta \cos[\xi] [m_{cc}^\nu \cos[\xi_\nu] + m_{sc}^\nu \sin[\xi_\nu]] + \beta \sin[\xi] [m_{cs}^\nu \cos[\xi_\nu] + m_{ss}^\nu \sin[\xi_\nu]]}}{\int d\xi e^{\beta \cos[\xi] [m_{cc}^\nu \cos[\xi_\nu] + m_{sc}^\nu \sin[\xi_\nu]] + \beta \sin[\xi] [m_{cs}^\nu \cos[\xi_\nu] + m_{ss}^\nu \sin[\xi_\nu]]}} \rangle_\xi, \quad (3.271)$$

The equilibrium values of the observables $\mathbf{m}_{**}$, as defined in (3.265,3.266), are now given by the solution of the coupled equations (3.268-3.271) which minimizes

$$f(\{\mathbf{m}_{**}\}) = \frac{1}{2} \sum_{**} \mathbf{m}_{**}^2 - \frac{1}{\beta} \langle \log \int d\xi \times e^{\beta \cos[\xi] [m_{cc}^\nu \cos[\xi_\nu] + m_{sc}^\nu \sin[\xi_\nu]] + \beta \sin[\xi] [m_{cs}^\nu \cos[\xi_\nu] + m_{ss}^\nu \sin[\xi_\nu]]} \rangle_\xi. \quad (3.272)$$

We can confirm that the relevant saddle-point must be a minimum by inspecting the $\beta = 0$ limit (infinite noise levels):

$$\lim_{\beta \rightarrow 0} f(\{\mathbf{m}_{**}\}) = \frac{1}{2} \sum_{**} \mathbf{m}_{**}^2 - \frac{1}{\beta} \log(2\pi).$$

From now on we will restrict our analysis to phase pattern components ξ_i^μ which have all been drawn independently at random from $[-\pi, \pi]$, with uniform probability density, so that $\langle g[\xi] \rangle_\xi = (2\pi)^{-p} \int_{-\pi}^\pi \dots \int_{-\pi}^\pi d\xi g[\xi]$. At $\beta = 0$ ($T = \infty$) one finds only the trivial state $m_{**}^\mu = 0$. It can be shown that there will be no discontinuous transitions to a non-trivial state as the noise level (temperature) is reduced. The continuous ones follow upon expansion of the equations (3.268-3.271) for small $\{\mathbf{m}_{**}\}$, which is found to give (for each μ and each combination $**$):

$$m_{**}^\mu = \frac{1}{4} \beta m_{**}^\mu + \mathcal{O}(\{\mathbf{m}_{**}^2\}).$$

Thus a continuous transition to recall states occurs at $T = \frac{1}{4}$. Full classification of all solutions of (3.268-3.271) is ruled out. Here we will restrict ourselves to the most relevant ones, such as the pure states, where $m_{**}^\mu = m_{**} \delta_{\mu\lambda}$ (for some pattern label λ). Here the oscillator phases are correlated with only one of the stored phase patterns (if at all). Insertion into the above expression for $f(\{\mathbf{m}_{**}\})$ shows that for such solutions we have to minimize [Coo01, SC00, SC01, CKS05]

$$f(\{m_{**}\}) = \frac{1}{2} \sum_{**} m_{**}^2 - \frac{1}{\beta} \int \frac{d\xi}{2\pi} \log \int d\xi \times e^{\beta \cos[\xi] [m_{cc} \cos[\xi] + m_{sc} \sin[\xi]] + \beta \sin[\xi] [m_{cs} \cos[\xi] + m_{ss} \sin[\xi]]}. \quad (3.273)$$

We anticipate solutions corresponding to the (partial) recall of the stored phase pattern ξ^λ or its mirror image (modulo overall phase shifts $\xi_i \rightarrow \xi_i + \delta$, under which the synapses are obviously invariant). Insertion into (3.268-3.271) of the state

$$\xi_i = \xi_i^\lambda + \delta \quad \text{gives} \quad (m_{cc}, m_{sc}, m_{cs}, m_{ss}) = \frac{1}{2}(\cos \delta, -\sin \delta, \sin \delta, \cos \delta).$$

Similarly, insertion into (3.268-3.271) of

$$\xi_i = -\xi_i^\lambda + \delta \quad \text{gives} \quad (m_{cc}, m_{sc}, m_{cs}, m_{ss}) = \frac{1}{2}(\cos \delta, \sin \delta, \sin \delta, -\cos \delta).$$

Thus we can identify retrieval states as those solutions which are of the form

$$\begin{aligned} (i) \text{ retrieval of } \boldsymbol{\xi}^\lambda : & \quad (m_{cc}, m_{sc}, m_{cs}, m_{ss}) = m(\cos \delta, -\sin \delta, \sin \delta, \cos \delta), \\ (ii) \text{ retrieval of } -\boldsymbol{\xi}^\lambda : & \quad (m_{cc}, m_{sc}, m_{cs}, m_{ss}) = m(\cos \delta, \sin \delta, \sin \delta, -\cos \delta), \end{aligned}$$

with full recall corresponding to $m = \frac{1}{2}$. Insertion into the saddle-point equations and into (3.273), followed by an appropriate shift of the integration variable ξ , shows that the free energy is independent of δ (so the above two ansätze solve the saddle-point equations for any δ) and that

$$m = \frac{1}{2} \frac{\int d\xi \cos[\xi] e^{\beta m \cos[\xi]}}{\int d\xi e^{\beta m \cos[\xi]}}, \quad f(m) = m^2 - \frac{1}{\beta} \log \int d\xi e^{\beta m \cos[\xi]}.$$

Expansion in powers of m , using $\log(1+z) = z - \frac{1}{2}z^2 + \mathcal{O}(z^3)$, reveals that non-zero minima m indeed bifurcate continuously at $T = \beta^{-1} = \frac{1}{4}$:

$$f(m) + \frac{1}{\beta} \log[2\pi] = (1 - \frac{1}{4}\beta)m^2 + \frac{1}{64}\beta^3 m^4 + \mathcal{O}(m^6). \quad (3.274)$$

Retrieval states are obviously not the only pure states that solve the saddle-point equations. The function (3.273) is invariant under the following discrete (non-commuting) transformations:

$$\begin{aligned} \text{I} : & \quad (m_{cc}, m_{sc}, m_{cs}, m_{ss}) \rightarrow (m_{cc}, m_{sc}, -m_{cs}, -m_{ss}), \\ \text{II} : & \quad (m_{cc}, m_{sc}, m_{cs}, m_{ss}) \rightarrow (m_{cs}, m_{ss}, m_{cc}, m_{sc}). \end{aligned}$$

We expect these to induce solutions with specific symmetries. In particular we anticipate the following symmetric and antisymmetric states:

$$\begin{aligned} & \text{symmetric under I} : (m_{cc}, m_{sc}, m_{cs}, m_{ss}) = \sqrt{2}m(\cos \delta, \sin \delta, 0, 0), \\ & \text{antisymmetric under I} : (m_{cc}, m_{sc}, m_{cs}, m_{ss}) = \sqrt{2}m(0, 0, \cos \delta, \sin \delta), \\ & \text{symmetric under II} : (m_{cc}, m_{sc}, m_{cs}, m_{ss}) = m(\cos \delta, \sin \delta, \cos \delta, \sin \delta), \\ & \text{antisymmetric under II} : (m_{cc}, m_{sc}, m_{cs}, m_{ss}) = m(\cos \delta, \sin \delta, -\cos \delta, -\sin \delta). \end{aligned}$$

Insertion into the saddle-point equations and into (3.273) shows in all four cases the parameter δ is arbitrary and that always

$$\begin{aligned} m &= \frac{1}{\sqrt{2}} \int \frac{d\xi}{2\pi} \cos[\xi] \frac{\int d\xi \cos[\xi] e^{\beta m \sqrt{2} \cos[\xi] \cos[\xi]}}{\int d\xi e^{\beta m \sqrt{2} \cos[\xi] \cos[\xi]}}, \\ f(m) &= m^2 - \frac{1}{\beta} \int \frac{d\xi}{2\pi} \log \int d\xi e^{\beta m \sqrt{2} \cos[\xi] \cos[\xi]}. \end{aligned}$$

Expansion in powers of m reveals that non-zero solutions m here again bifurcate continuously at $T = \frac{1}{4}$:

$$f(m) + \frac{1}{\beta} \log[2\pi] = (1 - \frac{1}{4}\beta)m^2 + \frac{3}{2} \cdot \frac{1}{64}\beta^3 m^4 + \mathcal{O}(m^6). \quad (3.275)$$

However, comparison with (3.274) shows that the free energy of the pure recall states is lower. Thus the system will prefer the recall states over the above solutions with specific symmetries.

Note, that the free energy and the order parameter equation for the pure recall states can be written in terms of *Bessel functions* as follows:

$$m = \frac{1}{2} \frac{I_1(\beta m)}{I_0(\beta m)}, \quad f(m) = m^2 - \frac{1}{\beta} \log[2\pi I_0(\beta m)]$$

3.3.6 Attractor Neural Nets with Binary Neurons

The simplest non-trivial recurrent neural networks consist of N binary neurons $\sigma_i \in \{-1, 1\}$, which respond stochastically to post-synaptic potentials (or local fields) $h_i(\boldsymbol{\sigma})$, with $\boldsymbol{\sigma} = (\sigma_1, \dots, \sigma_N)$. The fields depend linearly on the instantaneous neuron states,

$$h_i(\boldsymbol{\sigma}) = J_{ij}\sigma_j + \theta_i,$$

with the J_{ij} representing synaptic efficacies, and the θ_i representing external stimuli and/or neural thresholds [Coo01, SC00, SC01, CKS05].

Closed Macroscopic Laws for Sequential Dynamics

Here, we will first show how for sequential dynamics (where neurons are updated one after the other) one can calculate, from the microscopic stochastic laws, differential equations for the probability distribution of suitably defined macroscopic observables. For mathematical convenience our starting point will be the continuous-time master equation for the microscopic probability distribution $p_t(\boldsymbol{\sigma})$

$$\begin{aligned} \dot{p}_t(\boldsymbol{\sigma}) &= \{w_i(F_i\boldsymbol{\sigma})p_t(F_i\boldsymbol{\sigma}) - w_i(\boldsymbol{\sigma})p_t(\boldsymbol{\sigma})\}, \\ w_i(\boldsymbol{\sigma}) &= \frac{1}{2}[1 - \sigma_i \tanh[\beta h_i(\boldsymbol{\sigma})]], \end{aligned} \quad (3.276)$$

with $F_i\boldsymbol{\sigma} = \boldsymbol{\sigma}(\sigma_1, \dots, \sigma_{i-1}, -\sigma_i, \sigma_{i+1}, \dots, \sigma_N)$.

Let us illustrate the basic ideas with the help of a simple (infinite range) toy model: $J_{ij} = (J/N)\eta_i\xi_j$ and $\theta_i = 0$ (the variables η_i and ξ_i are arbitrary, but may not depend on N). For $\eta_i = \xi_i = 1$ we get a network with uniform synapses. For $\eta_i = \xi_i \in \{-1, 1\}$ and $J > 0$ we recover the Hopfield [Hop82] model with one stored pattern. Note: the synaptic matrix is non-symmetric

as soon as a pair (ij) exists such that $\eta_i \xi_j \neq \eta_j \xi_i$, so in general equilibrium statistical mechanics will not apply. The local fields become $h_i(\boldsymbol{\sigma}) = J \eta_i m(\boldsymbol{\sigma})$ with $m(\boldsymbol{\sigma}) = \frac{1}{N} \sum_k \xi_k \sigma_k$. Since they depend on the microscopic state $\boldsymbol{\sigma}$ only through the value of m , the latter quantity appears to constitute a natural macroscopic level of description. The probability density of finding the macroscopic state $m(\boldsymbol{\sigma}) = m$ is given by $\mathcal{P}_t[m] = \sum_{\boldsymbol{\sigma}} p_t(\boldsymbol{\sigma}) \delta[m - m(\boldsymbol{\sigma})]$. Its time derivative follows upon inserting (3.276):

$$\begin{aligned} \dot{\mathcal{P}}_t[m] &= \sum_{\boldsymbol{\sigma}} p_t(\boldsymbol{\sigma}) w_k(\boldsymbol{\sigma}) \left\{ \delta[m - m(\boldsymbol{\sigma}) + \frac{2}{N} \xi_k \sigma_k] - \delta[m - m(\boldsymbol{\sigma})] \right\} \\ &= \frac{\partial}{\partial m} \left\{ \sum_{\boldsymbol{\sigma}} p_t(\boldsymbol{\sigma}) \delta[m - m(\boldsymbol{\sigma})] \frac{2}{N} \xi_k \sigma_k w_k(\boldsymbol{\sigma}) \right\} + \mathcal{O}(\frac{1}{N}). \end{aligned}$$

Inserting our expressions for the transition rates $w_i(\boldsymbol{\sigma})$ and the local fields $h_i(\boldsymbol{\sigma})$ gives:

$$\dot{\mathcal{P}}_t[m] = \frac{\partial}{\partial m} \left\{ \mathcal{P}_t[m] \left[m - \frac{1}{N} \xi_k \tanh[\eta_k \beta J m] \right] \right\} + \mathcal{O}(N^{-1}).$$

In the limit $N \rightarrow \infty$ only the first term survives. The general solution of the resulting Liouville equation is

$$\mathcal{P}_t[m] = \int dm_0 \mathcal{P}_0[m_0] \delta[m - m(t|m_0)],$$

where $m(t|m_0)$ is the solution of

$$\dot{m} = \lim_{N \rightarrow \infty} \frac{1}{N} \xi_k \tanh[\eta_k \beta J m] - m, \quad m(0) = m_0. \quad (3.277)$$

This describes deterministic evolution; the only uncertainty in the value of m is due to uncertainty in initial conditions. If at $t = 0$ the quantity m is known exactly, this will remain the case for finite time-scales; m turns out to evolve in time according to (3.277).

Let us now allow for less trivial choices of the synaptic matrix $\{J_{ij}\}$ and try to calculate the evolution in time of a given set of macroscopic observables $\boldsymbol{\Omega}(\boldsymbol{\sigma}) = (\Omega_1(\boldsymbol{\sigma}), \dots, \Omega_n(\boldsymbol{\sigma}))$ in the limit $N \rightarrow \infty$. There are no restrictions yet on the form or the number n of these state variables; these will, however, arise naturally if we require the observables $\boldsymbol{\Omega}$ to obey a closed set of deterministic laws, as we will see. The probability density of finding the system in macroscopic state $\boldsymbol{\Omega}$ is given by [Coo01, SC00, SC01, CKS05]:

$$\mathcal{P}_t[\boldsymbol{\Omega}] = \sum_{\boldsymbol{\sigma}} p_t(\boldsymbol{\sigma}) \delta[\boldsymbol{\Omega} - \boldsymbol{\Omega}(\boldsymbol{\sigma})]. \quad (3.278)$$

Its time derivative is got by inserting (3.276). If in those parts of the resulting expression which contain the operators F_i we perform the transformations $\boldsymbol{\sigma} \rightarrow F_i \boldsymbol{\sigma}$, we arrive at

$$\dot{\mathcal{P}}_t[\Omega] = \sum_{\sigma} p_t(\sigma) w_i(\sigma) \{ \delta[\Omega - \Omega(F_i\sigma)] - \delta[\Omega - \Omega(\sigma)] \}.$$

If we define

$$\Omega_{\mu}(F_i\sigma) = \Omega_{\mu}(\sigma) + \Delta_{i\mu}(\sigma)$$

and make a Taylor expansion in powers of $\{\Delta_{i\mu}(\sigma)\}$, we finally get the so-called *Kramers–Moyal expansion*:⁹

$$\dot{\mathcal{P}}_t[\Omega] = \sum_{\ell \geq 1} \frac{(-1)^\ell}{\ell!} \sum_{\mu_1=1}^n \cdots \sum_{\mu_\ell=1}^n \frac{\partial^\ell}{\partial \Omega_{\mu_1} \cdots \partial \Omega_{\mu_\ell}} \left\{ \mathcal{P}_t[\Omega] F_{\mu_1 \dots \mu_\ell}^{(\ell)}[\Omega; t] \right\}. \quad (3.279)$$

It involves conditional averages $\langle f(\sigma) \rangle_{\Omega; t}$ and the ‘discrete derivatives’

$$\begin{aligned} \Delta_{j\mu}(\sigma) &= \Omega_{\mu}(F_j\sigma) - \Omega_{\mu}(\sigma), \\ F_{\mu_1 \dots \mu_\ell}^{(\ell)}[\Omega; t] &= \langle w_j(\sigma) \Delta_{j\mu_1}(\sigma) \cdots \Delta_{j\mu_\ell}(\sigma) \rangle_{\Omega; t}, \\ \langle f(\sigma) \rangle_{\Omega; t} &= \frac{\sum_{\sigma} p_t(\sigma) \delta[\Omega - \Omega(\sigma)] f(\sigma)}{\sum_{\sigma} p_t(\sigma) \delta[\Omega - \Omega(\sigma)]}. \end{aligned} \quad (3.280)$$

Retaining only the $\ell = 1$ term in (3.279) would lead us to a Liouville equation, which describes deterministic flow in Ω space. Including also the $\ell = 2$ term leads us to a Fokker–Planck equation which, in addition to flow, describes diffusion of the macroscopic probability density. Thus a sufficient condition for the observables $\Omega(\sigma)$ to evolve in time deterministically in the limit $N \rightarrow \infty$ is:

$$\lim_{N \rightarrow \infty} \sum_{\ell \geq 2} \frac{1}{\ell!} \sum_{\mu_1=1}^n \cdots \sum_{\mu_\ell=1}^n \langle |\Delta_{j\mu_1}(\sigma) \cdots \Delta_{j\mu_\ell}(\sigma)| \rangle_{\Omega; t} = 0. \quad (3.281)$$

In the simple case where all observables Ω_{μ} scale similarly in the sense that all ‘derivatives’ $\Delta_{j\mu} = \Omega_{\mu}(F_j\sigma) - \Omega_{\mu}(\sigma)$ are of the same order in N (i.e., there is a monotonic function $\tilde{\Delta}_N$ such that $\Delta_{j\mu} = \mathcal{O}(\tilde{\Delta}_N)$ for all $j\mu$), for instance, criterion (3.281) becomes:

$$\lim_{N \rightarrow \infty} n \tilde{\Delta}_N \sqrt{N} = 0. \quad (3.282)$$

If for a given set of observables condition (3.281) is satisfied, we can for large N describe the evolution of the macroscopic probability density by a Liouville equation:

⁹ The Kramers–Moyal expansion (3.279) is to be interpreted in a distributional sense, i.e., only to be used in expressions of the form $\int d\Omega_t(\Omega) G(\Omega)$ with smooth functions $G(\Omega)$, so that all derivatives are well-defined and finite. Furthermore, (3.279) will only be useful if the $\Delta_{j\mu}$, which measure the sensitivity of the macroscopic quantities to single neuron state changes, are sufficiently small. This is to be expected: for finite N any observable can only assume a finite number of possible values; only for $N \rightarrow \infty$ may we expect smooth probability distributions for our macroscopic quantities.

$$\dot{\mathcal{P}}_t[\Omega] = -\frac{\partial}{\partial \Omega_\mu} \left\{ \mathcal{P}_t[\Omega] F_\mu^{(1)}[\Omega; t] \right\},$$

whose solution describes deterministic flow,

$$\mathcal{P}_t[\Omega] = \int d\Omega_0 \mathcal{P}_0[\Omega_0] \delta[\Omega - \Omega(t|\Omega_0)],$$

with $\Omega(t|\Omega_0)$ given, in turn, as the solution of

$$\dot{\Omega}(t) = \mathbf{F}^{(1)}[\Omega(t); t], \quad \Omega(0) = \Omega_0. \quad (3.283)$$

In taking the limit $N \rightarrow \infty$, however, we have to keep in mind that the resulting deterministic theory is got by taking this limit for *finite* t . According to (3.279) the $\ell > 1$ terms do come into play for sufficiently large times t ; for $N \rightarrow \infty$, however, these times diverge by virtue of (3.281).

Now, equation (3.283) will in general not be autonomous; tracing back the origin of the explicit time dependence in the right-hand side of (3.283) one finds that to calculate $\mathbf{F}^{(1)}$ one needs to know the microscopic probability density $p_t(\sigma)$. This, in turn, requires solving equation (3.276) (which is exactly what one tries to avoid). We will now discuss a mechanism via which to eliminate the offending explicit time dependence, and to turn the observables $\Omega(\sigma)$ into an autonomous level of description, governed by *closed* dynamic laws. The idea is to choose the observables $\Omega(\sigma)$ in such a way that there is no explicit time dependence in the flow field $\mathbf{F}^{(1)}[\Omega; t]$ (if possible). According to (3.280) this implies making sure that there exist functions $\Phi_\mu[\Omega]$ such that

$$\lim_{N \rightarrow \infty} w_j(\sigma) \Delta_{j\mu}(\sigma) = \Phi_\mu[\Omega(\sigma)], \quad (3.284)$$

in which case the time dependence of $\mathbf{F}^{(1)}$ indeed drops out and the macroscopic state vector simply evolves in time according to [Coo01, SC00, SC01, CKS05]:

$$\dot{\Omega} = \Phi[\Omega], \quad \Phi = (\Phi_1[\Omega], \dots, \Phi_n[\Omega]).$$

Clearly, for this closure method to apply, a suitable separable structure of the synaptic matrix is required. If, for instance, the macroscopic observables Ω_μ depend linearly on the microscopic state variables σ (i.e., $\Omega_\mu(\sigma) = \frac{1}{N} \sum_{j=1}^N \omega_{\mu j} \sigma_j$), we get with the transition rates defined in (3.276):

$$\dot{\Omega}_\mu = \lim_{N \rightarrow \infty} \frac{1}{N} \omega_{\mu j} \tanh(\beta h_j(\sigma)) - \Omega_\mu, \quad (3.285)$$

in which case the only further condition for (3.284) to hold is that all local fields $h_k(\sigma)$ must (in leading order in N) depend on the microscopic state σ only through the values of the observables Ω ; since the local fields depend linearly on σ this, in turn, implies that the synaptic matrix must be separable: if $J_{ij} = \sum_\mu K_{i\mu} \omega_{\mu j}$ then indeed $h_i(\sigma) = \sum_\mu K_{i\mu} \Omega_\mu(\sigma) + \theta_i$. Next we will show how this approach can be applied to networks for which the

matrix of synapses has a separable form (which includes most symmetric and non-symmetric Hebbian type attractor models). We will restrict ourselves to models with $\theta_i = 0$; introducing non-zero thresholds is straightforward and does not pose new problems.

Application to Separable Attractor Networks

We consider the following class of models, in which the interaction matrices have the form

$$J_{ij} = \frac{1}{N} Q(\xi_i; \xi_j), \quad \xi_i = (\xi_i^1, \dots, \xi_i^p). \quad (3.286)$$

The components ξ_i^μ , representing the information ('patterns') to be stored or processed, are assumed to be drawn from a finite discrete set Λ , containing n_Λ elements (they are not allowed to depend on N). The Hopfield model [Hop82] corresponds to choosing $Q(\mathbf{x}; \mathbf{y}) = \mathbf{x} \cdot \mathbf{y}$ and $\Lambda \equiv \{-1, 1\}$. One now introduces a partition of the system $\{1, \dots, N\}$ into n_Λ^p so-called sublattices I_η :

$$I_\eta = \{i \mid \xi_i = \eta\}, \quad \{1, \dots, N\} = \bigcup_{\eta} I_\eta, \quad (\eta \in \Lambda^p).$$

The number of neurons in sublattice I_η is denoted by $|I_\eta|$ (this number will have to be large). If we choose as our macroscopic observables the average activities ('magnetizations') within these sublattices, we are able to express the local fields h_k solely in terms of macroscopic quantities:

$$m_\eta(\sigma) = \frac{1}{|I_\eta|} \sum_{i \in I_\eta} \sigma_i, \quad h_k(\sigma) = p_\eta Q(\xi_k; \eta) m_\eta, \quad (3.287)$$

with the relative sublattice sizes $p_\eta = |I_\eta|/N$. If all p_η are of the same order in N (which, for example, is the case if the vectors ξ_i have been drawn at random from the set Λ^p) we may write $\Delta_{j\eta} = \mathcal{O}(n_\Lambda^p N^{-1})$ and use (3.282). The evolution in time of the sublattice activities is then found to be deterministic in the $N \rightarrow \infty$ limit if $\lim_{N \rightarrow \infty} p/\log N = 0$. Furthermore, condition (3.284) holds, since

$$w_j(\sigma) \Delta_{j\eta}(\sigma) = \tanh[\beta p_{\eta'} Q(\eta; \eta') m_{\eta'}] - m_\eta.$$

This situation is described by (3.285), and that the evolution in time of the sublattice activities is governed by the following autonomous set of differential equations [RKH88]:

$$\dot{m}_\eta = \tanh[\beta p_{\eta'} Q(\eta; \eta') m_{\eta'}] - m_\eta. \quad (3.288)$$

We see that, in contrast to the equilibrium techniques as described above, here there is no need at all to require symmetry of the interaction matrix or absence of self-interactions. In the symmetric case $Q(\mathbf{x}; \mathbf{y}) = Q(\mathbf{y}; \mathbf{x})$ the

system will approach equilibrium; if the kernel Q is positive definite this can be shown, for instance, by inspection of the *Lyapunov function* $\mathcal{L}\{m_\eta\}$:¹⁰

$$\begin{aligned}\mathcal{L}\{m_\eta\} &= \frac{1}{2} p_\eta m_\eta Q(\eta; \eta') m_{\eta'} p_{\eta'} \\ &\quad - \frac{1}{\beta} \sum_\eta p_\eta \log \cosh[\beta Q(\eta; \eta') m_{\eta'} p_{\eta'}],\end{aligned}$$

which is bounded from below and obeys:

$$\dot{\mathcal{L}} = -[p_\eta \dot{m}_\eta] Q(\eta; \eta') [p_{\eta'} \dot{m}_{\eta'}] \leq 0. \quad (3.289)$$

Note that from the sublattice activities, in turn, follow the *Hopfield overlaps* $m_\mu(\sigma)$,

$$m_\mu(\sigma) = \frac{1}{N} \xi_i^\mu \sigma_i = p_\eta \eta_\mu m_\eta. \quad (3.290)$$

Simple examples of relevant models of the type (3.286), the dynamics of which are for large N described by equation (3.288), are for instance the ones where one applies a nonlinear operation Φ to the standard Hopfield-type [Hop82] (or Hebbian) interactions. This nonlinearity could result from e.g., a clipping procedure or from retaining only the *sign* of the Hebbian values:

$$\begin{aligned}J_{ij} &= \frac{1}{N} \Phi(\xi_i^\mu \xi_j^\mu) : \quad \text{e.g.,} \\ \Phi(x) &= \begin{cases} -K & \text{for } x \leq -K \\ x & \text{for } -K < x < K \\ K & \text{for } x \geq K \end{cases}, \quad \text{or} \quad \Phi(x) = \text{sgn}(x).\end{aligned}$$

The effect of introducing such nonlinearities is found to be of a quantitative nature, giving rise to little more than a re-scaling of critical noise levels and storage capacities. We will illustrate this statement by working out the $p = 2$ equations for randomly drawn pattern bits $\xi_i^\mu \in \{-1, 1\}$, where there are only four sub-lattices, and where $p_\eta = \frac{1}{4}$ for all η (details can be found in e.g., [DHS91]). Using $\Phi(0) = 0$ and $\Phi(-x) = -\Phi(x)$ (as with the above examples) we get from (3.288):

$$\dot{m}_\eta = \tanh\left[\frac{1}{4}\beta\Phi(2)(m_\eta - m_{-\eta})\right] - m_\eta. \quad (3.291)$$

Here the choice made for $\Phi(x)$ shows up only as a re-scaling of the temperature. From (3.291) we further get $\frac{d}{dt}(m_\eta + m_{-\eta}) = -(m_\eta + m_{-\eta})$. The system decays exponentially towards a state where, according to (3.290), $m_\eta = -m_{-\eta}$

¹⁰ Recall that *Lyapunov function* is a function of the state variables which is bounded from below and whose value decreases monotonically during the dynamics, see e.g., [Kha92]. Its existence guarantees evolution towards a stationary state (under some weak conditions).

for all $\boldsymbol{\eta}$. If at $t = 0$ this is already the case, we find (at least for $p = 2$) decoupled equations for the sub-lattice activities.

Now, equations (3.288, 3.290) suggest that at the level of overlaps there will be, in turn, closed laws if the kernel Q is bilinear, $Q(\mathbf{x}; \mathbf{y}) = \sum_{\mu\nu} x_\mu A_{\mu\nu} y_\nu$, or [Coo01, SC00, SC01, CKS05]:

$$J_{ij} = \frac{1}{N} \xi_i^\mu A_{\mu\nu} \xi_j^\nu, \quad \boldsymbol{\xi}_i = (\xi_i^1, \dots, \xi_i^p). \quad (3.292)$$

We will see that now the ξ_i^μ need not be drawn from a finite discrete set (as long as they do not depend on N). The Hopfield model corresponds to $A_{\mu\nu} = \delta_{\mu\nu}$ and $\xi_i^\mu \in \{-1, 1\}$. The fields h_k can now be written in terms of the overlaps m_μ :

$$h_k(\boldsymbol{\sigma}) = \boldsymbol{\xi}_k \cdot \mathbf{A} \mathbf{m}(\boldsymbol{\sigma}), \quad \mathbf{m} = (m_1, \dots, m_p), \quad m_\mu(\boldsymbol{\sigma}) = \frac{1}{N} \xi_i^\mu \sigma_i.$$

For this choice of macroscopic variables we find $\Delta_{j\mu} = \mathcal{O}(N^{-1})$, so the evolution of the vector $\mathbf{m}$ becomes deterministic for $N \rightarrow \infty$ if, according to (3.282), $\lim_{N \rightarrow \infty} p/\sqrt{N} = 0$. Again (3.284) holds, since

$$w_j(\boldsymbol{\sigma}) \Delta_{j\mu}(\boldsymbol{\sigma}) = \frac{1}{N} \boldsymbol{\xi}_k \tanh[\beta \boldsymbol{\xi}_k \cdot \mathbf{A} \mathbf{m}] - \mathbf{m}.$$

Thus the evolution in time of the overlap vector $\mathbf{m}$ is governed by a closed set of differential equations:

$$\dot{\mathbf{m}} = \langle \boldsymbol{\xi} \tanh[\beta \boldsymbol{\xi} \cdot \mathbf{A} \mathbf{m}] \rangle_{\boldsymbol{\xi}} - \mathbf{m}, \quad \langle \Phi(\boldsymbol{\xi}) \rangle_{\boldsymbol{\xi}} = \int d\boldsymbol{\xi} \rho(\boldsymbol{\xi}) \Phi(\boldsymbol{\xi}), \quad (3.293)$$

with $\rho(\boldsymbol{\xi}) = \lim_{N \rightarrow \infty} N^{-1} \sum_i \delta[\boldsymbol{\xi} - \boldsymbol{\xi}_i]$. Symmetry of the synapses is not required. For certain non-symmetric matrices A one finds stable limit-cycle solutions of (3.293). In the symmetric case $A_{\mu\nu} = A_{\nu\mu}$ the system will approach equilibrium; the Lyapunov function (3.289) for positive definite matrices A now becomes:

$$\mathcal{L}\{\mathbf{m}\} = \frac{1}{2} \mathbf{m} \cdot \mathbf{A} \mathbf{m} - \frac{1}{\beta} \langle \log \cosh[\beta \boldsymbol{\xi} \cdot \mathbf{A} \mathbf{m}] \rangle_{\boldsymbol{\xi}}.$$

As a second simple application of the flow equations (3.293) we turn to the relaxation times corresponding to the attractors of the Hopfield model (where $A_{\mu\nu} = \delta_{\mu\nu}$). Expanding (3.293) near a stable fixed-point $\mathbf{m}^*$, i.e., $\mathbf{m}(t) = \mathbf{m}^* + \mathbf{x}(t)$ with $|\mathbf{x}(t)| \ll 1$, gives the linearized equation

$$\dot{x}_\mu = [\beta \langle \xi_\mu \xi_\nu \tanh[\beta \boldsymbol{\xi} \cdot \mathbf{m}^*] \rangle_{\boldsymbol{\xi}} - \delta_{\mu\nu}] x_\nu + \mathcal{O}(\mathbf{x}^2). \quad (3.294)$$

The Jacobian of (3.293), which determines the linearized equation (3.294), turns out to be *minus* the curvature matrix of the free energy surface at the fixed-point. The asymptotic relaxation towards any stable attractor is generally exponential, with a characteristic time τ given by the inverse of the

smallest eigenvalue of the curvature matrix. If, in particular, for the fixed-point $\mathbf{m}^*$ we substitute an n -mixture state, i.e., $m_\mu = m_n$ ($\mu \leq n$) and $m_\mu = 0$ ($\mu > n$), and transform (3.294) to the basis where the corresponding curvature matrix $\mathbf{D}^{(n)}$ (with eigenvalues D_λ^n) is diagonal, $\mathbf{x} \rightarrow \tilde{\mathbf{x}}$, we get $\tilde{x}_\lambda(t) = \tilde{x}_\lambda(0)e^{-tD_\lambda^n} + \dots$ so $\tau^{-1} = \min_\lambda D_\lambda^n$, which we have already calculated in determining the character of the saddle-points of the free-energy surface. The relaxation time for the n -mixture attractors decreases monotonically with the degree of mixing n , for any noise level. At the transition where a macroscopic state $\mathbf{m}^*$ ceases to correspond to a local minimum of the free energy surface, it also de-stabilizes in terms of the linearized dynamic equation (3.294) (as it should). The Jacobian develops a zero eigenvalue, the relaxation time diverges, and the long-time behavior is no longer got from the linearized equation. This gives rise to critical slowing down (i.e., power law relaxation as opposed to exponential relaxation). For instance, at the transition temperature $T_c = 1$ for the $n = 1$ (pure) state, we find by expanding (3.293):

$$\dot{m}_\mu = m_\mu \left[\frac{2}{3} m_\mu^2 - \mathbf{m}^2 \right] + \mathcal{O}(\mathbf{m}^5),$$

which gives rise to a relaxation towards the trivial fixed-point of the form $\mathbf{m} \sim t^{-\frac{1}{2}}$.

If one is willing to restrict oneself to the limited class of models (3.292) (as opposed to the more general class (3.286)) and to the more global level of description in terms of p overlap parameters m_μ instead of n_A^p sublattice activities m_η , then there are two rewards. Firstly there will be no restrictions on the stored pattern components ξ_i^μ (for instance, they are allowed to be real-valued); secondly the number p of patterns stored can be much larger for the deterministic autonomous dynamical laws to hold ($p \ll \sqrt{N}$ instead of $p \ll \log N$, which from a biological point of view is not impressive [Coo01, SC00, SC01, CKS05]).

Closed Macroscopic Laws for Parallel Dynamics

We now turn to the parallel dynamics counterpart of (3.276), i.e., the Markov chain

$$p_{\ell+1}(\boldsymbol{\sigma}) = \sum_{\boldsymbol{\sigma}'} W[\boldsymbol{\sigma}; \boldsymbol{\sigma}'] p_\ell(\boldsymbol{\sigma}'), \quad (3.295)$$

$$W[\boldsymbol{\sigma}; \boldsymbol{\sigma}'] = \prod_{i=1}^N \frac{1}{2} [1 + \sigma_i \tanh[\beta h_i(\boldsymbol{\sigma}')]],$$

with $\sigma_i \in \{-1, 1\}$, and with local fields $h_i(\boldsymbol{\sigma})$ defined in the usual way. The evolution of macroscopic probability densities will here be described by discrete maps, instead of differential equations.

Let us first see what happens to our previous toy model: $J_{ij} = (J/N)\eta_i\xi_j$ and $\theta_i = 0$. As before we try to describe the dynamics at the (macroscopic)

level of the quantity $m(\boldsymbol{\sigma}) = \frac{1}{N} \sum_k \xi_k \sigma_k$. The evolution of the macroscopic probability density $\mathcal{P}_t[m]$ is got by inserting (3.295):

$$\begin{aligned} \mathcal{P}_{t+1}[m] &= \sum_{\boldsymbol{\sigma}\boldsymbol{\sigma}'} \delta[m - m(\boldsymbol{\sigma})] W[\boldsymbol{\sigma}; \boldsymbol{\sigma}'] p_t(\boldsymbol{\sigma}') \\ &= \int dm' \tilde{W}_t[m, m'] \mathcal{P}_t[m'], \quad \text{with} \\ \tilde{W}_t[m, m'] &= \frac{\sum_{\boldsymbol{\sigma}\boldsymbol{\sigma}'} \delta[m - m(\boldsymbol{\sigma})] \delta[m' - m(\boldsymbol{\sigma}')] W[\boldsymbol{\sigma}; \boldsymbol{\sigma}'] p_t(\boldsymbol{\sigma}')}{\sum_{\boldsymbol{\sigma}'} \delta[m' - m(\boldsymbol{\sigma}')] p_t(\boldsymbol{\sigma}')} \end{aligned} \quad (3.296)$$

We now insert our expression for the transition probabilities $W[\boldsymbol{\sigma}; \boldsymbol{\sigma}']$ and for the local fields. Since the fields depend on the microscopic state $\boldsymbol{\sigma}$ only through $m(\boldsymbol{\sigma})$, the distribution $p_t(\boldsymbol{\sigma})$ drops out of the above expression for $\tilde{W}_t$ which thereby loses its explicit time-dependence, $\tilde{W}_t[m, m'] \rightarrow \tilde{W}[m, m']$:

$$\begin{aligned} \tilde{W}[m, m'] &= e^{-\sum_i \log \cosh(\beta J m' \eta_i)} \langle \delta[m - m(\boldsymbol{\sigma})] e^{\beta J m' \eta_i \sigma_i} \rangle_{\boldsymbol{\sigma}} \\ \text{with} \quad \langle \dots \rangle_{\boldsymbol{\sigma}} &= 2^{-N} \sum_{\boldsymbol{\sigma}} \dots \end{aligned}$$

Inserting the integral representation for the δ -function allows us to perform the average:

$$\tilde{W}[m, m'] = \left[\frac{\beta N}{2\pi} \right] \int dk e^{N\Psi(m, m', k)},$$

$$\Psi = i\beta k m + \langle \log \cosh \beta[J\eta m' - ik\xi] \rangle_{\eta, \xi} - \langle \log \cosh \beta[J\eta m'] \rangle_{\eta}.$$

Since $\tilde{W}[m, m']$ is (by construction) normalized, $\int dm \tilde{W}[m, m'] = 1$, we find that for $N \rightarrow \infty$ the expectation value with respect to $\tilde{W}[m, m']$ of any sufficiently smooth function $f(m)$ will be determined only by the value $m^*(m')$ of m in the relevant saddle-point of Ψ :

$$\int dm f(m) \tilde{W}[m, m'] = \frac{\int dm dk f(m) e^{N\Psi(m, m', k)}}{\int dm dk e^{N\Psi(m, m', k)}} \rightarrow f(m^*(m')), \quad (N \rightarrow \infty).$$

Variation of Ψ with respect to k and m gives the two saddle-point equations:

$$m = \langle \xi \tanh \beta[J\eta m' - \xi k] \rangle_{\eta, \xi}, \quad k = 0.$$

We may now conclude that $\lim_{N \rightarrow \infty} \tilde{W}[m, m'] = \delta[m - m^*(m')]$ with $m^*(m') = \langle \xi \tanh(\beta J \eta m') \rangle_{\eta, \xi}$, and that the macroscopic equation (3.296) becomes [Coo01, SC00, SC01, CKS05]:

$$\mathcal{P}_{t+1}[m] = \int dm' \delta[m - \langle \xi \tanh(\beta J \eta m') \rangle_{\eta, \xi}] \mathcal{P}_t[m'], \quad (N \rightarrow \infty).$$

This describes deterministic evolution. If at $t = 0$ we know m exactly, this will remain the case for finite time-scales, and m will evolve according to a discrete version of the sequential dynamics law (3.277):

$$m_{t+1} = \langle \xi \tanh[\beta J \eta m_t] \rangle_{\eta, \xi}.$$

We now try to generalize the above approach to less trivial classes of models. As for the sequential case we will find in the limit $N \rightarrow \infty$ closed deterministic evolution equations for a more general set of intensive macroscopic state variables $\mathbf{\Omega}(\boldsymbol{\sigma}) = (\Omega_1(\boldsymbol{\sigma}), \dots, \Omega_n(\boldsymbol{\sigma}))$ if the local fields $h_i(\boldsymbol{\sigma})$ depend on the microscopic state $\boldsymbol{\sigma}$ only through the values of $\mathbf{\Omega}(\boldsymbol{\sigma})$, and if the number n of these state variables necessary to do so is not too large. The evolution of the ensemble probability density (3.278) is now got by inserting the Markov equation (3.295):

$$\mathcal{P}_{t+1}[\mathbf{\Omega}] = \int d\mathbf{\Omega}' \tilde{W}_t[\mathbf{\Omega}, \mathbf{\Omega}'] \mathcal{P}_t[\mathbf{\Omega}'], \quad (3.297)$$

$$\begin{aligned} \tilde{W}_t[\mathbf{\Omega}, \mathbf{\Omega}'] &= \frac{\sum_{\boldsymbol{\sigma}, \boldsymbol{\sigma}'} \delta[\mathbf{\Omega} - \mathbf{\Omega}(\boldsymbol{\sigma})] \delta[\mathbf{\Omega}' - \mathbf{\Omega}(\boldsymbol{\sigma}')] W[\boldsymbol{\sigma}; \boldsymbol{\sigma}'] p_t(\boldsymbol{\sigma}')}{\sum_{\boldsymbol{\sigma}'} \delta[\mathbf{\Omega}' - \mathbf{\Omega}(\boldsymbol{\sigma}')] p_t(\boldsymbol{\sigma}')} \quad (3.298) \\ &= \langle \delta[\mathbf{\Omega} - \mathbf{\Omega}(\boldsymbol{\sigma})] \langle e^{[\beta \sigma_i h_i(\boldsymbol{\sigma}') - \log \cosh(\beta h_i(\boldsymbol{\sigma}'))]} \rangle_{\mathbf{\Omega}', t} \rangle_{\boldsymbol{\sigma}}, \end{aligned}$$

with $\langle \dots \rangle_{\boldsymbol{\sigma}} = 2^{-N} \sum_{\boldsymbol{\sigma}} \dots$, and with the conditional (or sub-shell) average defined as in (3.280). It is clear from (3.298) that in order to find autonomous macroscopic laws, i.e., for the distribution $p_t(\boldsymbol{\sigma})$ to drop out, the local fields must depend on the microscopic state $\boldsymbol{\sigma}$ only through the macroscopic quantities $\mathbf{\Omega}(\boldsymbol{\sigma})$: $h_i(\boldsymbol{\sigma}) = h_i[\mathbf{\Omega}(\boldsymbol{\sigma})]$. In this case $\tilde{W}_t$ loses its explicit time-dependence, $\tilde{W}_t[\mathbf{\Omega}, \mathbf{\Omega}'] \rightarrow \tilde{W}[\mathbf{\Omega}, \mathbf{\Omega}']$. Inserting integral representations for the δ -functions leads to:

$$\begin{aligned} \tilde{W}[\mathbf{\Omega}, \mathbf{\Omega}'] &= \left[\frac{\beta N}{2\pi} \right]^n \int d\mathbf{K} e^{N\Psi(\mathbf{\Omega}, \mathbf{\Omega}', \mathbf{K})}, \\ \Psi &= i\beta \mathbf{K} \cdot \mathbf{\Omega} + \frac{1}{N} \log \langle e^{\beta [\sum_i \sigma_i h_i[\mathbf{\Omega}'] - iN\mathbf{K} \cdot \mathbf{\Omega}(\boldsymbol{\sigma})]} \rangle_{\boldsymbol{\sigma}} \\ &\quad - \frac{1}{N} \sum_i \log \cosh[\beta h_i[\mathbf{\Omega}']]. \end{aligned}$$

Using the normalization $\int d\mathbf{\Omega} \tilde{W}[\mathbf{\Omega}, \mathbf{\Omega}'] = 1$, we can write expectation values with respect to $\tilde{W}[\mathbf{\Omega}, \mathbf{\Omega}']$ of macroscopic quantities $f[\mathbf{\Omega}]$ as

$$\int d\mathbf{\Omega} f[\mathbf{\Omega}] \tilde{W}[\mathbf{\Omega}, \mathbf{\Omega}'] = \frac{\int d\mathbf{\Omega} d\mathbf{K} f[\mathbf{\Omega}] e^{N\Psi(\mathbf{\Omega}, \mathbf{\Omega}', \mathbf{K})}}{\int d\mathbf{\Omega} d\mathbf{K} e^{N\Psi(\mathbf{\Omega}, \mathbf{\Omega}', \mathbf{K})}}. \quad (3.299)$$

For saddle-point arguments to apply in determining the leading order in N of (3.299), we encounter restrictions on the number n of our macroscopic quantities (as expected), since n determines the dimension of the integrations in (3.299). The restrictions can be found by expanding Ψ around its maximum Ψ^* . After defining $\mathbf{x} = (\mathbf{\Omega}, \mathbf{K})$, of dimension $2n$, and after translating the location of the maximum to the origin, one has [Coo01, SC00, SC01, CKS05]

$$\Psi(\mathbf{x}) = \Psi^* - \frac{1}{2}x_\mu x_\nu H_{\mu\nu} + x_\mu x_\nu x_\rho L_{\mu\nu\rho} + \mathcal{O}(\mathbf{x}^4), \quad \text{giving}$$

$$\begin{aligned} \frac{\int d\mathbf{x} g(\mathbf{x}) e^{N\Psi(\mathbf{x})}}{\int d\mathbf{x} g(\mathbf{x}) e^{N\Psi(\mathbf{x})}} - g(\mathbf{0}) &= \frac{\int d\mathbf{x} [g(\mathbf{x}) - g(\mathbf{0})] e^{-\frac{1}{2}N\mathbf{x}\cdot\mathbf{H}\mathbf{x} + Nx_\mu x_\nu x_\rho L_{\mu\nu\rho} + \mathcal{O}(N\mathbf{x}^4)}}{\int d\mathbf{x} e^{-\frac{1}{2}N\mathbf{x}\cdot\mathbf{H}\mathbf{x} + Nx_\mu x_\nu x_\rho L_{\mu\nu\rho} + \mathcal{O}(N\mathbf{x}^4)}} \\ &= \frac{\int d\mathbf{y} [g(\mathbf{y}/\sqrt{N}) - g(\mathbf{0})] e^{-\frac{1}{2}\mathbf{y}\cdot\mathbf{H}\mathbf{y} + y_\mu y_\nu y_\rho L_{\mu\nu\rho}/\sqrt{N} + \mathcal{O}(\mathbf{y}^4/N)}}{\int d\mathbf{y} e^{-\frac{1}{2}\mathbf{y}\cdot\mathbf{H}\mathbf{y} + y_\mu y_\nu y_\rho L_{\mu\nu\rho}/\sqrt{N} + \mathcal{O}(\mathbf{y}^4/N)}} = \\ &= \frac{\int d\mathbf{y} \left[N^{-\frac{1}{2}}\mathbf{y} \cdot \nabla g(\mathbf{0}) + \mathcal{O}(\mathbf{y}^2/N) \right] e^{-\frac{1}{2}\mathbf{y}\cdot\mathbf{H}\mathbf{y}} \left[1 + y_\mu y_\nu y_\rho L_{\mu\nu\rho}/\sqrt{N} + \mathcal{O}(\mathbf{y}^6/N) \right]}{\int d\mathbf{y} e^{-\frac{1}{2}\mathbf{y}\cdot\mathbf{H}\mathbf{y}} \left[1 + y_\mu y_\nu y_\rho L_{\mu\nu\rho}/\sqrt{N} + \mathcal{O}(\mathbf{y}^6/N) \right]} \\ &= \mathcal{O}(n^2/N) + \mathcal{O}(n^4/N^2) + \text{non-dominant terms}, \quad (N, n \rightarrow \infty), \end{aligned}$$

with $\mathbf{H}$ denoting the Hessian (curvature) matrix of the surface Ψ at the minimum Ψ^* . We thus find

$$\lim_{N \rightarrow \infty} n/\sqrt{N} = 0 : \quad \lim_{N \rightarrow \infty} \int d\mathbf{\Omega} f[\mathbf{\Omega}] \tilde{W}[\mathbf{\Omega}, \mathbf{\Omega}'] = f[\mathbf{\Omega}^*(\mathbf{\Omega}')],$$

where $\mathbf{\Omega}^*(\mathbf{\Omega}')$ denotes the value of $\mathbf{\Omega}$ in the saddle-point where Ψ is minimized. Variation of Ψ with respect to $\mathbf{\Omega}$ and $\mathbf{K}$ gives the saddle-point equations:

$$\mathbf{\Omega} = \frac{\langle \mathbf{\Omega}(\boldsymbol{\sigma}) e^{\beta[\sigma_i h_i[\mathbf{\Omega}'] - iN\mathbf{K}\cdot\mathbf{\Omega}(\boldsymbol{\sigma})]} \rangle_{\boldsymbol{\sigma}}}{\langle e^{\beta[\sigma_i h_i[\mathbf{\Omega}'] - iN\mathbf{K}\cdot\mathbf{\Omega}(\boldsymbol{\sigma})]} \rangle_{\boldsymbol{\sigma}}}, \quad \mathbf{K} = 0.$$

We may now conclude that $\lim_{N \rightarrow \infty} \tilde{W}[\mathbf{\Omega}, \mathbf{\Omega}'] = \delta[\mathbf{\Omega} - \mathbf{\Omega}^*(\mathbf{\Omega}')]$, with

$$\mathbf{\Omega}^*(\mathbf{\Omega}') = \frac{\langle \mathbf{\Omega}(\boldsymbol{\sigma}) e^{\beta\sigma_i h_i[\mathbf{\Omega}']} \rangle_{\boldsymbol{\sigma}}}{\langle e^{\beta\sigma_i h_i[\mathbf{\Omega}']} \rangle_{\boldsymbol{\sigma}}},$$

and that for $N \rightarrow \infty$ the macroscopic equation (3.297) becomes $\mathcal{P}_{t+1}[\mathbf{\Omega}] = \int d\mathbf{\Omega}' \delta[\mathbf{\Omega} - \mathbf{\Omega}^*(\mathbf{\Omega}')] \mathcal{P}_t[\mathbf{\Omega}']$. This relation again describes deterministic evolution. If at $t = 0$ we know $\mathbf{\Omega}$ exactly, this will remain the case for finite time-scales and $\mathbf{\Omega}$ will evolve according to

$$\mathbf{\Omega}(t+1) = \frac{\langle \mathbf{\Omega}(\boldsymbol{\sigma}) e^{\beta\sigma_i h_i[\mathbf{\Omega}(t)]} \rangle_{\boldsymbol{\sigma}}}{\langle e^{\beta\sigma_i h_i[\mathbf{\Omega}(t)]} \rangle_{\boldsymbol{\sigma}}}. \quad (3.300)$$

As with the sequential case, in taking the limit $N \rightarrow \infty$ we have to keep in mind that the resulting laws apply to finite t , and that for sufficiently large times terms of higher order in N do come into play. As for the sequential case, a more rigorous and tedious analysis shows that the restriction $n/\sqrt{N} \rightarrow 0$ can in fact be weakened to $n/N \rightarrow 0$. Finally, for macroscopic quantities $\mathbf{\Omega}(\boldsymbol{\sigma})$ which are linear in $\boldsymbol{\sigma}$, the remaining $\boldsymbol{\sigma}$ -averages become trivial, so that [Ber91]:

$$\Omega_\mu(\boldsymbol{\sigma}) = \frac{1}{N} \omega_{\mu i} \sigma_i : \quad \Omega_\mu(t+1) = \lim_{N \rightarrow \infty} \frac{1}{N} \omega_{\mu i} \tanh[\beta h_i[\mathbf{\Omega}(t)]] .$$

Application to Separable Attractor Networks

The separable attractor models (3.286), described at the level of sublattice activities (3.287), indeed have the property that all local fields can be written in terms of the macroscopic observables. What remains to ensure deterministic evolution is meeting the condition on the number of sublattices. If all relative sublattice sizes $p_{\boldsymbol{\eta}}$ are of the same order in N (as for randomly drawn patterns) this condition again translates into $\lim_{N \rightarrow \infty} p / \log N = 0$ (as for sequential dynamics). Since the sublattice activities are linear functions of the σ_i , their evolution in time is governed by equation (3.300), which acquires the form:

$$m_{\boldsymbol{\eta}}(t+1) = \tanh[\beta p_{\boldsymbol{\eta}'} Q(\boldsymbol{\eta}; \boldsymbol{\eta}') m_{\boldsymbol{\eta}'}(t)]. \quad (3.301)$$

As for sequential dynamics, symmetry of the interaction matrix does not play a role.

At the more global level of overlaps $m_{\mu}(\boldsymbol{\sigma}) = N^{-1} \sum_i \xi_i^{\mu} \sigma_i$ we, in turn, get autonomous deterministic laws if the local fields $h_i(\boldsymbol{\sigma})$ can be expressed in terms of $\mathbf{m}(\boldsymbol{\sigma})$ only, as for the models (3.292) (or, more generally, for all models in which the interactions are of the form $J_{ij} = \sum_{\mu \leq p} f_{i\mu} \xi_j^{\mu}$), and with the following restriction on the number p of embedded patterns: $\lim_{N \rightarrow \infty} p / \sqrt{N} = 0$ (as with sequential dynamics). For the bilinear models (3.292), the evolution in time of the overlap vector $\mathbf{m}$ (which depends linearly on the σ_i) is governed by (3.300), which now translates into the iterative map:

$$\mathbf{m}(t+1) = \langle \boldsymbol{\xi} \tanh[\beta \boldsymbol{\xi} \cdot \mathbf{A} \mathbf{m}(t)] \rangle_{\boldsymbol{\xi}}, \quad (3.302)$$

with $\rho(\boldsymbol{\xi})$ as defined in (3.293). Again symmetry of the synapses is not required. For parallel dynamics it is far more difficult than for sequential dynamics to construct Lyapunov functions, and prove that the macroscopic laws (3.302) for symmetric systems evolve towards a stable fixed-point (as one would expect), but it can still be done. For non-symmetric systems the macroscopic laws (3.302) can in principle display all the interesting, but complicated, phenomena of non-conservative nonlinear systems. Nevertheless, it is also not uncommon that the equations (3.302) for non-symmetric systems can be mapped by a time-dependent transformation onto the equations for related symmetric systems (mostly variants of the original Hopfield model).

Note that the fixed-points of the macroscopic equations (3.288) and (3.293) (derived for sequential dynamics) are identical to those of (3.301) and (3.302) (derived for parallel dynamics). The stability properties of these fixed-points, however, need not be the same, and have to be assessed on a case-by-case basis. For the Hopfield model, i.e., equations (3.293, 3.302) with $A_{\mu\nu} = \delta_{\mu\nu}$, they are found to be the same, but already for $A_{\mu\nu} = -\delta_{\mu\nu}$ the two types of dynamics would behave differently [Coo01, SC00, SC01, CKS05].

3.3.7 Attractor Neural Nets with Continuous Neurons

Closed Macroscopic Laws

We have seen above that models of recurrent neural networks with continuous neural variables (e.g., graded-response neurons or coupled oscillators) can often be described by a Fokker-Planck equation for the microscopic state probability density $p_t(\boldsymbol{\sigma})$:

$$\dot{p}_t(\boldsymbol{\sigma}) = -\frac{\partial}{\partial \sigma_i} [p_t(\boldsymbol{\sigma}) f_i(\boldsymbol{\sigma})] + T \sum_i \frac{\partial^2}{\partial \sigma_i^2} p_t(\boldsymbol{\sigma}).$$

Averages over $p_t(\boldsymbol{\sigma})$ are denoted by $\langle G \rangle = \int d\boldsymbol{\sigma} p_t(\boldsymbol{\sigma}) G(\boldsymbol{\sigma}, t)$. From (3.221) one gets directly (through integration by parts) an equation for the time derivative of averages [Coo01, SC00, SC01, CKS05]:

$$\frac{d}{dt} \langle G \rangle = \left\langle \frac{\partial G}{\partial t} \right\rangle + \left\langle \left[f_i(\boldsymbol{\sigma}) + T \frac{\partial}{\partial \sigma_i} \right] \frac{\partial G}{\partial \sigma_i} \right\rangle. \quad (3.303)$$

In particular, if we apply (3.303) to $G(\boldsymbol{\sigma}, t) = \delta[\boldsymbol{\Omega} - \boldsymbol{\Omega}(\boldsymbol{\sigma})]$, for any set of macroscopic observables $\boldsymbol{\Omega}(\boldsymbol{\sigma}) = (\Omega_1(\boldsymbol{\sigma}), \dots, \Omega_n(\boldsymbol{\sigma}))$ (in the spirit of the previous section), we get a dynamic equation for the macroscopic probability density $P_t(\boldsymbol{\Omega}) = \langle \delta[\boldsymbol{\Omega} - \boldsymbol{\Omega}(\boldsymbol{\sigma})] \rangle$, which is again of the Fokker-Planck form:

$$\begin{aligned} \dot{P}_t(\boldsymbol{\Omega}) = & -\frac{\partial}{\partial \Omega_\mu} \left\{ P_t(\boldsymbol{\Omega}) \left\langle \left[f_i(\boldsymbol{\sigma}) + T \frac{\partial}{\partial \sigma_i} \right] \frac{\partial}{\partial \sigma_i} \Omega_\mu(\boldsymbol{\sigma}) \right\rangle_{\boldsymbol{\Omega};t} \right\} \\ & + T \frac{\partial^2}{\partial \Omega_\mu \partial \Omega_\nu} \left\{ P_t(\boldsymbol{\Omega}) \left\langle \left[\frac{\partial}{\partial \sigma_i} \Omega_\mu(\boldsymbol{\sigma}) \right] \left[\frac{\partial}{\partial \sigma_i} \Omega_\nu(\boldsymbol{\sigma}) \right] \right\rangle_{\boldsymbol{\Omega};t} \right\}, \end{aligned} \quad (3.304)$$

with the conditional (or sub-shell) averages:

$$\langle G(\boldsymbol{\sigma}) \rangle_{\boldsymbol{\Omega},t} = \frac{\int d\boldsymbol{\sigma} p_t(\boldsymbol{\sigma}) \delta[\boldsymbol{\Omega} - \boldsymbol{\Omega}(\boldsymbol{\sigma})] G(\boldsymbol{\sigma})}{\int d\boldsymbol{\sigma} p_t(\boldsymbol{\sigma}) \delta[\boldsymbol{\Omega} - \boldsymbol{\Omega}(\boldsymbol{\sigma})]}.$$

From (3.304) we infer that a sufficient condition for the observables $\boldsymbol{\Omega}(\boldsymbol{\sigma})$ to evolve in time deterministically (i.e., for having vanishing diffusion matrix elements in (3.304)) in the limit $N \rightarrow \infty$ is

$$\lim_{N \rightarrow \infty} \left\langle \sum_i \left[\sum_\mu \left| \frac{\partial}{\partial \sigma_i} \Omega_\mu(\boldsymbol{\sigma}) \right| \right]^2 \right\rangle_{\boldsymbol{\Omega};t} = 0. \quad (3.305)$$

If (3.305) holds, the macroscopic Fokker-Planck equation (3.304) reduces for $N \rightarrow \infty$ to a Liouville equation, and the observables $\boldsymbol{\Omega}(\boldsymbol{\sigma})$ will evolve in time according to the coupled deterministic equations:

$$\dot{\Omega}_\mu = \lim_{N \rightarrow \infty} \left\langle \left[f_i(\boldsymbol{\sigma}) + T \frac{\partial}{\partial \sigma_i} \right] \frac{\partial}{\partial \sigma_i} \Omega_\mu(\boldsymbol{\sigma}) \right\rangle_{\boldsymbol{\Omega};t}. \quad (3.306)$$

The deterministic macroscopic equation (3.306), together with its associated condition for validity (3.305) will form the basis for the subsequent analysis.

The general derivation given above went smoothly. However, the equations (3.306) are not yet closed. It turns out that to achieve closure even for simple continuous networks we can no longer get away with just a finite (small) number of macroscopic observables (as with binary neurons). This we will now illustrate with a simple toy network of graded-response neurons:

$$\dot{u}_i(t) = J_{ij} g[u_j(t)] - u_i(t) + \eta_i(t),$$

with $g[z] = \frac{1}{2}[\tanh(\gamma z) + 1]$ and with the standard Gaussian white noise $\eta_i(t)$. In the language of (3.221) this means

$$f_i(\mathbf{u}) = J_{ij} g[u_j] - u_i.$$

We choose uniform synapses $J_{ij} = J/N$, so $f_i(\mathbf{u}) \rightarrow (J/N) \sum_j g[u_j] - u_i$. If (3.305) were to hold, we would find the deterministic macroscopic laws

$$\dot{\Omega}_\mu = \lim_{N \rightarrow \infty} \left\langle \sum_i \left[\frac{J}{N} \sum_j g[u_j] - u_i + T \frac{\partial}{\partial u_i} \right] \frac{\partial}{\partial u_i} \Omega_\mu(\mathbf{u}) \right\rangle_{\Omega, t}. \quad (3.307)$$

In contrast to similar models with binary neurons, choosing as our macroscopic level of description $\Omega(\mathbf{u})$ again simply the average $m(\mathbf{u}) = N^{-1} \sum_i u_i$ now leads to an equation which fails to close [Coo01, SC00, SC01, CKS05]:

$$\dot{m} = \lim_{N \rightarrow \infty} J \left\langle \frac{1}{N} \sum_j g[u_j] \right\rangle_{m, t} - m.$$

The term $N^{-1} \sum_j g[u_j]$ cannot be written as a function of $N^{-1} \sum_i u_i$. We might be tempted to try dealing with this problem by just including the offending term in our macroscopic set, and choose $\Omega(\mathbf{u}) = (N^{-1} \sum_i u_i, N^{-1} \sum_i g[u_i])$. This would indeed solve our closure problem for the m -equation, but we would now find a new closure problem in the equation for the newly introduced observable. The only way out is to choose an observable *function*, namely the distribution of potentials

$$\rho(u; \mathbf{u}) = \frac{1}{N} \sum_i \delta[u - u_i], \quad (3.308)$$

$$\rho(u) = \langle \rho(u; \mathbf{u}) \rangle = \left\langle \frac{1}{N} \sum_i \delta[u - u_i] \right\rangle.$$

This is to be done with care, in view of our restriction on the number of observables: we evaluate (3.308) at first only for n specific values u_μ and take the limit $n \rightarrow \infty$ only after the limit $N \rightarrow \infty$. Thus we define $\Omega_\mu(\mathbf{u}) = \frac{1}{N} \sum_i \delta[u_\mu - u_i]$, condition (3.305) reduces to the familiar expression $\lim_{N \rightarrow \infty} n/\sqrt{N} = 0$, and we get for $N \rightarrow \infty$ and $n \rightarrow \infty$ (taken in that

order) from (3.307) a diffusion equation for the distribution of membrane potentials (describing a so-called ‘*time-dependent Ornstein-Uhlenbeck process*’ [Kam92, Gar94]):

$$\dot{\rho}(u) = -\frac{\partial}{\partial u} \left\{ \rho(u) \left[J \int du' \rho(u') g[u'] - u \right] \right\} + T \frac{\partial^2}{\partial u^2} \rho(u). \quad (3.309)$$

The natural solution of (3.309)¹¹ is the Gaussian distribution

$$\rho_t(u) = [2\pi\Sigma^2(t)]^{-\frac{1}{2}} e^{-\frac{1}{2}[u-\bar{u}(t)]^2/\Sigma^2(t)}, \quad (3.310)$$

in which $\Sigma = [T + (\Sigma_0^2 - T)e^{-2t}]^{\frac{1}{2}}$, and $\bar{u}$ evolves in time according to

$$\frac{d}{dt}\bar{u} = J \int Dz g[\bar{u} + \Sigma z] - \bar{u},$$

with $Dz = (2\pi)^{-\frac{1}{2}} e^{-\frac{1}{2}z^2} dz$. We can now also calculate the distribution $p(s)$ of neuronal firing activities $s_i = g[u_i]$ at any time,

$$p(s) = \int du \rho(u) \delta[s - g[u]] = \frac{\rho(g^{\text{inv}}[s])}{\int_0^1 ds' \rho(g^{\text{inv}}[s'])}.$$

For our choice $g[z] = \frac{1}{2} + \frac{1}{2} \tanh[\gamma z]$ we have $g^{\text{inv}}[s] = \frac{1}{2\gamma} \log[s/(1-s)]$, so in combination with (3.310)¹²

$$0 < s < 1 : \quad p(s) = \frac{e^{-\frac{1}{2}[(2\gamma)^{-1} \log[s/(1-s)] - \bar{u}]^2/\Sigma^2}}{\int_0^1 ds' e^{-\frac{1}{2}[(2\gamma)^{-1} \log[s'/(1-s')] - \bar{u}]^2/\Sigma^2}}.$$

Application to Graded-Response Attractor Networks

We will now turn to attractor networks with graded-response neurons of the type (3.222), in which p binary patterns $\xi^\mu = (\xi_1^\mu, \dots, \xi_N^\mu) \in \{-1, 1\}^N$ have been stored via separable Hebbian synapses (3.292): $J_{ij} = (2/N) \xi_i^\mu A_{\mu\nu} \xi_j^\nu$ (the extra factor 2 is inserted for future convenience). Adding suitable thresholds $\theta_i = -\frac{1}{2} \sum_j J_{ij}$ to the right-hand sides of (3.222), and choosing the nonlinearity $g(z) = \frac{1}{2}(1 + \tanh[\gamma z])$ would then give us [Coo01, SC00, SC01, CKS05]

¹¹ For non-Gaussian initial conditions $\rho_0(u)$ the solution of (3.309) would in time converge towards the Gaussian solution.

¹² None of the above results (not even those on the stationary state) could have been got within equilibrium statistical mechanics, since any network of connected graded-response neurons will violate detailed balance. Secondly, there appears to be a qualitative difference between simple networks (e.g., $J_{ij} = J/N$) of binary neurons versus those of continuous neurons, in terms of the types of macroscopic observables needed for deriving closed deterministic laws: a single number $m = N^{-1} \sum_i \sigma_i$ versus a distribution $\rho(\sigma) = N^{-1} \sum_i \delta[\sigma - \sigma_i]$.

$$\dot{u}_i(t) = \sum_{\mu\nu} \xi_i^\mu A_{\mu\nu} \frac{1}{N} \sum_j \xi_j^\nu \tanh[\gamma u_j(t)] - u_i(t) + \eta_i(t),$$

so the deterministic forces are $f_i(\mathbf{u}) = N^{-1} \sum_{\mu\nu} \xi_i^\mu A_{\mu\nu} \sum_j \xi_j^\nu \tanh[\gamma u_j] - u_i$. Choosing our macroscopic observables $\Omega(\mathbf{u})$ such that (3.305) holds, would lead to the deterministic macroscopic laws

$$\begin{aligned} \dot{\Omega}_\mu &= \lim_{N \rightarrow \infty} \sum_{\mu\nu} A_{\mu\nu} \left\langle \left[\frac{1}{N} \xi_j^\nu \tanh[\gamma u_j] \right] \left[\xi_i^\mu \frac{\partial}{\partial u_i} \Omega_\mu(\mathbf{u}) \right] \right\rangle_{\Omega; t} \\ &+ \lim_{N \rightarrow \infty} \left\langle \left[T \frac{\partial}{\partial u_i} - u_i \right] \frac{\partial}{\partial u_i} \Omega_\mu(\mathbf{u}) \right\rangle_{\Omega; t}. \end{aligned} \quad (3.311)$$

As with the uniform synapses case, the main problem to be dealt with is how to choose the $\Omega_\mu(\mathbf{u})$ such that (3.311) closes. It turns out that the canonical choice is to turn to the distributions of membrane potentials within each of the 2^p sub-lattices, as introduced above :

$$I_\eta = \{i \mid \xi_i = \eta\} : \quad \rho_\eta(u; \mathbf{u}) = \frac{1}{|I_\eta|} \sum_{i \in I_\eta} \delta[u - u_i], \quad (3.312)$$

$$\rho_\eta(u) = \langle \rho_\eta(u; \mathbf{u}) \rangle,$$

with $\eta \in \{-1, 1\}^p$ and $\lim_{N \rightarrow \infty} |I_\eta|/N = p_\eta$. Again we evaluate the distributions in (3.312) at first only for n specific values u_μ and send $n \rightarrow \infty$ after $N \rightarrow \infty$. Now condition (3.305) reduces to $\lim_{N \rightarrow \infty} 2^p / \sqrt{N} = 0$. We will keep p finite, for simplicity. Using identities such as $\sum_i \dots = \sum_\eta \sum_{i \in I_\eta} \dots$ and $i \in I_\eta$:

$$\frac{\partial}{\partial u_i} \rho_\eta(u; \mathbf{u}) = -|I_\eta|^{-1} \frac{\partial}{\partial u} \delta[u - u_i], \quad \frac{\partial^2}{\partial u_i^2} \rho_\eta(u; \mathbf{u}) = |I_\eta|^{-1} \frac{\partial^2}{\partial u^2} \delta[u - u_i],$$

we then get for $N \rightarrow \infty$ and $n \rightarrow \infty$ (taken in that order) from equation (3.311) 2^p coupled diffusion equations for the distributions $\rho_\eta(u)$ of membrane potentials in each of the 2^p sub-lattices I_η :

$$\dot{\rho}_\eta(u) = -\frac{\partial}{\partial u} \{ \rho_\eta(u) [\eta_\mu A_{\mu\nu} p_{\eta'} \eta'_\nu \int du' \rho_{\eta'}(u') \tanh[\gamma u'] - u] \} + T \frac{\partial^2}{\partial u^2} \rho_\eta(u). \quad (3.313)$$

Equation (3.313) is the basis for our further analysis. It can be simplified only if we make additional assumptions on the system's initial conditions, such as δ -distributed or Gaussian distributed $\rho_\eta(u)$ at $t = 0$ (see below); otherwise it will have to be solved numerically.

It is clear that (3.313) is again of the time-dependent Ornstein-Uhlenbeck form, and will thus again have Gaussian solutions as the natural ones [Coo01, SC00, SC01, CKS05]:

$$\rho_{t,\eta}(u) = [2\pi \Sigma_\eta^2(t)]^{-\frac{1}{2}} e^{-\frac{1}{2} [u - \bar{u}_\eta(t)]^2 / \Sigma_\eta^2(t)},$$

in which $\Sigma_{\boldsymbol{\eta}}(t) = [T + (\Sigma_{\boldsymbol{\eta}}^2(0) - T)e^{-2t}]^{\frac{1}{2}}$, and with the $\bar{u}_{\boldsymbol{\eta}}(t)$ evolving in time according to

$$\frac{d}{dt}\bar{u}_{\boldsymbol{\eta}} = p_{\boldsymbol{\eta}'}(\boldsymbol{\eta} \cdot \mathbf{A}\boldsymbol{\eta}') \int Dz \tanh[\gamma(\bar{u}_{\boldsymbol{\eta}'} + \Sigma_{\boldsymbol{\eta}'} z)] - \bar{u}_{\boldsymbol{\eta}}. \quad (3.314)$$

Our problem has thus been reduced successfully to the study of the 2^p coupled scalar equations (3.314). We can also measure the correlation between the firing activities $s_i(u_i) = \frac{1}{2}[1 + \tanh(\gamma u_i)]$ and the pattern components (similar to the overlaps in the case of binary neurons). If the pattern bits are drawn at random, i.e., $\lim_{N \rightarrow \infty} |I_{\boldsymbol{\eta}}|/N = p_{\boldsymbol{\eta}} = 2^{-p}$ for all $\boldsymbol{\eta}$, we can define a ‘graded-response’ equivalent $m_{\mu}(\mathbf{u}) = 2N^{-1}\xi_i^{\mu} s_i(u_i) \in [-1, 1]$ of the Hopfield pattern overlaps:

$$\begin{aligned} m_{\mu}(\mathbf{u}) &= \frac{2}{N}\xi_i^{\mu} s_i(\mathbf{u}) = \frac{1}{N}\xi_i^{\mu} \tanh(\gamma u_i) + \mathcal{O}(N^{-\frac{1}{2}}) \\ &= p_{\boldsymbol{\eta}} \eta_{\mu} \int du \rho_{\boldsymbol{\eta}}(u; \mathbf{u}) \tanh(\gamma u) + \mathcal{O}(N^{-\frac{1}{2}}). \end{aligned}$$

Full recall of pattern μ implies $s_i(u_i) = \frac{1}{2}[\xi_i^{\mu} + 1]$, giving $m_{\mu}(\mathbf{u}) = 1$. Since the distributions $\rho_{\boldsymbol{\eta}}(u)$ obey deterministic laws for $N \rightarrow \infty$, the same will be true for the overlaps $\mathbf{m} = (m_1, \dots, m_p)$. For the Gaussian solutions (3.314) of (3.313) we can now proceed to replace the 2^p macroscopic laws (3.314), which reduce to $\frac{d}{dt}\bar{u}_{\boldsymbol{\eta}} = \boldsymbol{\eta} \cdot \mathbf{A}\mathbf{m} - \bar{u}_{\boldsymbol{\eta}}$ and give $\bar{u}_{\boldsymbol{\eta}} = \bar{u}_{\boldsymbol{\eta}}(0)e^{-t} + \boldsymbol{\eta} \cdot \mathbf{A} \int_0^t ds e^{s-t}\mathbf{m}(s)$, by p integral equations in terms of overlaps only:

$$\begin{aligned} m_{\mu}(t) &= p_{\boldsymbol{\eta}} \eta_{\mu} \int Dz \tanh[\gamma(\bar{u}_{\boldsymbol{\eta}}(0)e^{-t} \\ &\quad + \boldsymbol{\eta} \cdot \mathbf{A} \int_0^t ds e^{s-t}\mathbf{m}(s) + z\sqrt{T + (\Sigma_{\boldsymbol{\eta}}^2(0) - T)e^{-2t}})], \end{aligned} \quad (3.315)$$

with $Dz = (2\pi)^{-\frac{1}{2}}e^{-\frac{1}{2}z^2}dz$. Here the sub-lattices only come in via the initial conditions.

The equations describing the asymptotic (stationary) state can be written entirely without sub-lattices,¹³

$$\begin{aligned} m_{\mu} &= \langle \xi_{\mu} \int Dz \tanh[\gamma(\boldsymbol{\xi} \cdot \mathbf{A}\mathbf{m} + z\sqrt{T})] \rangle_{\boldsymbol{\xi}}, \\ \rho_{\boldsymbol{\eta}}(u) &= [2\pi T]^{-\frac{1}{2}} e^{-\frac{1}{2}[u - \boldsymbol{\eta} \cdot \mathbf{A}\mathbf{m}]^2/T}, \end{aligned} \quad (3.316)$$

by taking the $t \rightarrow \infty$ limit in (3.315), using $\bar{u}_{\boldsymbol{\eta}} \rightarrow \boldsymbol{\eta} \cdot \mathbf{A}\mathbf{m}$, $\Sigma_{\boldsymbol{\eta}} \rightarrow \sqrt{T}$, and the familiar notation

¹³ Note the appealing similarity with previous results on networks with binary neurons in equilibrium. For $T = 0$ the overlap equations (3.316) become identical to those found for attractor networks with binary neurons and finite p (hence our choice to insert an extra factor 2 in defining the synapses), with γ replacing the inverse noise level β in the former.

$$\langle g(\xi) \rangle_{\xi} = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_i g(\xi_i) = 2^{-p} \sum_{\xi \in \{-1,1\}^p} g(\xi).$$

For the simplest non-trivial choice, $A_{\mu\nu} = \delta_{\mu\nu}$ (i.e., $J_{ij} = (2/N) \sum_{\mu} \xi_i^{\mu} \xi_j^{\mu}$, as in the Hopfield [Hop82] model) equation (3.316) yields the familiar pure and mixture state solutions. For $T = 0$ we find a continuous phase transition from non-recall to pure states of the form $m_{\mu} = m \delta_{\mu\nu}$ (for some ν) at $\gamma_c = 1$. For $T > 0$ we have in (3.316) an additional Gaussian noise, absent in the models with binary neurons. Again the pure states are the first non-trivial solutions to enter the stage. Substituting $m_{\mu} = m \delta_{\mu\nu}$ into (3.316) gives

$$m = \int Dz \tanh[\gamma(m + z\sqrt{T})]. \quad (3.317)$$

Writing (3.317) as $m^2 = \gamma m \int_0^m dk [1 - \int Dz \tanh^2[\gamma(k + z\sqrt{T})]] \leq \gamma m^2$, reveals that $m = 0$ as soon as $\gamma < 1$. A continuous transition to an $m > 0$ state occurs when

$$\gamma^{-1} = 1 - \int Dz \tanh^2[\gamma z \sqrt{T}].$$

A parametrization of this transition line in the (γ, T) -plane is given by

$$\begin{aligned} \gamma^{-1}(x) &= 1 - \int Dz \tanh^2(zx), \\ T(x) &= x^2 / \gamma^2(x), \quad x \geq 0. \end{aligned}$$

Discontinuous transitions away from $m = 0$ (for which there is no evidence) would have to be calculated numerically. For $\gamma = \infty$ we get the equation $m = \text{erf}[m/\sqrt{2T}]$, giving a continuous transition to $m > 0$ at $T_c = 2/\pi \approx 0.637$. Alternatively the latter number can also be found by taking $\lim_{x \rightarrow \infty} T(x)$ in the above parametrization:

$$\begin{aligned} T_c(\gamma = \infty) &= \lim_{x \rightarrow \infty} x^2 [1 - \int Dz \tanh^2(zx)]^2 \\ &= \lim_{x \rightarrow \infty} \left[\int Dz \frac{\partial}{\partial z} \tanh(zx) \right]^2 = [2 \int Dz \delta(z)]^2 = 2/\pi. \end{aligned}$$

Let us now turn to dynamics. It follows from (3.316) that the ‘natural’ initial conditions for $\bar{u}_{\boldsymbol{\eta}}$ and $\Sigma_{\boldsymbol{\eta}}$ are of the form: $\bar{u}_{\boldsymbol{\eta}}(0) = \boldsymbol{\eta} \cdot \mathbf{k}_0$ and $\Sigma_{\boldsymbol{\eta}}(0) = \Sigma_0$ for all $\boldsymbol{\eta}$. Equivalently:

$$\begin{aligned} t = 0 : \quad \rho_{\boldsymbol{\eta}}(u) &= [2\pi \Sigma_0]^{-\frac{1}{2}} e^{-\frac{1}{2}[u - \boldsymbol{\eta} \cdot \mathbf{k}_0]^2 / \Sigma_0}, \\ \mathbf{k}_0 &\in \mathbb{R}^p, \quad \Sigma_0 \in \mathbb{R}. \end{aligned}$$

These would also be the typical and natural statistics if we were to prepare an initial firing state $\{s_i\}$ by hand, via manipulation of the potentials $\{u_i\}$.

For such initial conditions we can simplify the dynamical equation (3.315) to [Coo01, SC00, SC01, CKS05]

$$m_\mu(t) = \langle \xi_\mu \int Dz \tanh[\gamma(\xi \cdot [\mathbf{k}_0 e^{-t} + \mathbf{A} \int_0^t ds e^{s-t} \mathbf{m}(s)] + z\sqrt{T + (\Sigma_0^2 - T)e^{-2t}})] \rangle_{\xi}. \quad (3.318)$$

$$+ \mathbf{A} \int_0^t ds e^{s-t} \mathbf{m}(s)] + z\sqrt{T + (\Sigma_0^2 - T)e^{-2t}})]_{\xi}. \quad (3.319)$$

For the special case of the Hopfield synapses, i.e., $A_{\mu\nu} = \delta_{\mu\nu}$, it follows from (3.318) that recall of a given pattern ν is triggered upon choosing $k_{0,\mu} = k_0 \delta_{\mu\nu}$ (with $k_0 > 0$), since then equation (3.318) generates $m_\mu(t) = m(t) \delta_{\mu\nu}$ at any time, with the amplitude $m(t)$ following from

$$m(t) = \int Dz \tanh[\gamma[k_0 e^{-t} + \int_0^t ds e^{s-t} m(s) + z\sqrt{T + (\Sigma_0^2 - T)e^{-2t}}]], \quad (3.320)$$

which is the dynamical counterpart of equation (3.317) (to which indeed it reduces for $t \rightarrow \infty$).

We finally specialize further to the case where our Gaussian initial conditions are not only chosen to trigger recall of a single pattern ξ^ν , but in addition describe uniform membrane potentials within the sub-lattices, i.e., $k_{0,\mu} = k_0 \delta_{\mu\nu}$ and $\Sigma_0 = 0$, so $\rho_{\mathbf{\eta}}(u) = \delta[u - k_0 \eta_\nu]$. Here we can derive from (3.320) at $t = 0$ the identity $m_0 = \tanh[\gamma k_0]$, which enables us to express k_0 as $k_0 = (2\gamma)^{-1} \log[(1 + m_0)/(1 - m_0)]$, and find (3.320) reducing to

$$m(t) = \int Dz \tanh[e^{-t} \log[\frac{1 + m_0}{1 - m_0}]^{\frac{1}{2}} + \gamma[\int_0^t ds e^{s-t} m(s) + z\sqrt{T(1 - e^{-2t})}]].$$

Compared to the overlap evolution in large networks of binary networks (away from saturation) one can see richer behavior, e.g., non-monotonicity [Coo01, SC00, SC01, CKS05].

3.3.8 Correlation- and Response-Functions

We now turn to correlation functions $C_{ij}(t, t')$ and response functions $G_{ij}(t, t')$. These will become the language in which the partition function methods are formulated, which will enable us to solve the dynamics of recurrent networks in the (complex) regime near saturation (we take $t > t'$):

$$C_{ij}(t, t') = \langle \sigma_i(t) \sigma_j(t') \rangle, \quad G_{ij}(t, t') = \partial \langle \sigma_i(t) \rangle / \partial \theta_j(t') \quad (3.321)$$

The $\{\sigma_i\}$ evolve in time according to equations of the form (3.276) (binary neurons, sequential updates), (3.295) (binary neurons, parallel updates) or (3.221) (continuous neurons). The θ_i represent thresholds and/or external stimuli, which are added to the local fields in the cases (3.276, 3.295), or added to the deterministic forces in the case of a Fokker-Planck equation (3.221).

We retain $\theta_i(t) = \theta_i$, except for a perturbation $\delta\theta_j(t')$ applied at time t' in defining the response function. Calculating averages such as (3.321) requires determining joint probability distributions involving neuron states at different times.

Fluctuation–Dissipation Theorems

For networks of binary neurons with discrete time dynamics of the form $p_{\ell+1}(\boldsymbol{\sigma}) = \sum_{\boldsymbol{\sigma}'} W[\boldsymbol{\sigma}; \boldsymbol{\sigma}'] p_{\ell}(\boldsymbol{\sigma}')$, the probability of observing a given ‘path’ $\boldsymbol{\sigma}(\ell') \rightarrow \boldsymbol{\sigma}(\ell' + 1) \rightarrow \dots \rightarrow \boldsymbol{\sigma}(\ell - 1) \rightarrow \boldsymbol{\sigma}(\ell)$ of successive configurations between step ℓ' and step ℓ is given by the product of the corresponding transition matrix elements (without summation):

$$\text{Prob}[\boldsymbol{\sigma}(\ell'), \dots, \boldsymbol{\sigma}(\ell)] = W[\boldsymbol{\sigma}(\ell); \boldsymbol{\sigma}(\ell - 1)] W[\boldsymbol{\sigma}(\ell - 1); \boldsymbol{\sigma}(\ell - 2)] \dots W[\boldsymbol{\sigma}(\ell' + 1); \boldsymbol{\sigma}(\ell')] p_{\ell'}(\boldsymbol{\sigma}(\ell')).$$

This allows us to write [Coo01, SC00, SC01, CKS05]

$$\begin{aligned} C_{ij}(\ell, \ell') &= \sum_{\boldsymbol{\sigma}(\ell')} \dots \sum_{\boldsymbol{\sigma}(\ell)} \text{Prob}[\boldsymbol{\sigma}(\ell'), \dots, \boldsymbol{\sigma}(\ell)] \sigma_i(\ell) \sigma_j(\ell') \quad (3.322) \\ &= \sum_{\boldsymbol{\sigma} \boldsymbol{\sigma}'} \sigma_i \sigma'_j W^{\ell - \ell'}[\boldsymbol{\sigma}; \boldsymbol{\sigma}'] p_{\ell'}(\boldsymbol{\sigma}'), \end{aligned}$$

$$G_{ij}(\ell, \ell') = \sum_{\boldsymbol{\sigma} \boldsymbol{\sigma}' \boldsymbol{\sigma}''} \sigma_i W^{\ell - \ell' - 1}[\boldsymbol{\sigma}; \boldsymbol{\sigma}''] \left[\frac{\partial}{\partial \theta_j} W[\boldsymbol{\sigma}''; \boldsymbol{\sigma}'] \right] p_{\ell'}(\boldsymbol{\sigma}'). \quad (3.323)$$

From (3.322) and (3.323) it follows that both $C_{ij}(\ell, \ell')$ and $G_{ij}(\ell, \ell')$ will in the stationary state, i.e., upon substituting $p_{\ell'}(\boldsymbol{\sigma}') = p_{\infty}(\boldsymbol{\sigma}')$, only depend on $\ell - \ell'$: $C_{ij}(\ell, \ell') \rightarrow C_{ij}(\ell - \ell')$ and $G_{ij}(\ell, \ell') \rightarrow G_{ij}(\ell - \ell')$. For this we do not require detailed balance. Detailed balance, however, leads to a simple relation between the response function $G_{ij}(\tau)$ and the temporal derivative of the correlation function $C_{ij}(\tau)$.

We now turn to equilibrium systems, i.e., networks with symmetric synapses (and with all $J_{ii} = 0$ in the case of sequential dynamics). We calculate the derivative of the transition matrix that occurs in (3.323) by differentiating the equilibrium condition $p_{\text{eq}}(\boldsymbol{\sigma}) = \sum_{\boldsymbol{\sigma}'} W[\boldsymbol{\sigma}; \boldsymbol{\sigma}'] p_{\text{eq}}(\boldsymbol{\sigma}')$ with respect to external fields:

$$\frac{\partial}{\partial \theta_j} p_{\text{eq}}(\boldsymbol{\sigma}) = \sum_{\boldsymbol{\sigma}'} \left\{ \frac{\partial W[\boldsymbol{\sigma}; \boldsymbol{\sigma}']}{\partial \theta_j} p_{\text{eq}}(\boldsymbol{\sigma}') + W[\boldsymbol{\sigma}; \boldsymbol{\sigma}'] \frac{\partial}{\partial \theta_j} p_{\text{eq}}(\boldsymbol{\sigma}') \right\}.$$

Detailed balance implies $p_{\text{eq}}(\boldsymbol{\sigma}) = Z^{-1} e^{-\beta H(\boldsymbol{\sigma})}$ (in the parallel case we simply substitute the appropriate Hamiltonian $H \rightarrow \tilde{H}$), giving $\partial p_{\text{eq}}(\boldsymbol{\sigma}) / \partial \theta_j = -[Z^{-1} \partial Z / \partial \theta_j + \beta \partial H(\boldsymbol{\sigma}) / \partial \theta_j] p_{\text{eq}}(\boldsymbol{\sigma})$, so that

$$\sum_{\sigma'} \frac{\partial W[\sigma; \sigma']}{\partial \theta_j} p_{\text{eq}}(\sigma') = \beta \left\{ \sum_{\sigma'} W[\sigma; \sigma'] \frac{\partial H(\sigma')}{\partial \theta_j} p_{\text{eq}}(\sigma') - \frac{\partial H(\sigma)}{\partial \theta_j} p_{\text{eq}}(\sigma) \right\},$$

in which the term containing Z drops out. We now get for the response function (3.323) in equilibrium the following result:

$$G_{ij}(\ell) = \beta \sum_{\sigma \sigma'} \sigma_i W^{\ell-1}[\sigma; \sigma'] \left\{ \sum_{\sigma''} W[\sigma'; \sigma''] \frac{\partial H(\sigma'')}{\partial \theta_j} p_{\text{eq}}(\sigma'') - \frac{\partial H(\sigma')}{\partial \theta_j} p_{\text{eq}}(\sigma') \right\}. \quad (3.324)$$

The structure of (3.324) is similar to what follows upon calculating the evolution of the equilibrium correlation function (3.322) in a single iteration step:

$$C_{ij}(\ell) - C_{ij}(\ell-1) = \sum_{\sigma \sigma'} \sigma_i W^{\ell-1}[\sigma; \sigma'] \times \left\{ \sum_{\sigma''} W[\sigma'; \sigma''] \sigma_j'' p_{\text{eq}}(\sigma'') - \sigma_j' p_{\text{eq}}(\sigma') \right\}. \quad (3.325)$$

Finally we calculate the relevant derivatives of the two Hamiltonians

$$H(\sigma) = -J_{ij} \sigma_i \sigma_j + \theta_i \sigma_i, \quad \text{and} \\ \tilde{H}(\sigma) = -\theta_i \sigma_i - \beta^{-1} \sum_i \log 2 \cosh[\beta h_i(\sigma)]$$

(with $h_i(\sigma) = J_{ij} \sigma_j + \theta_i$),

$$\frac{\partial H(\sigma)}{\partial \theta_j} = -\sigma_j, \quad \frac{\partial \tilde{H}(\sigma)}{\partial \theta_j} = -\sigma_j - \tanh[\beta h_j(\sigma)].$$

For sequential dynamics we hereby arrive directly at a *fluctuation-dissipation theorem*. For parallel dynamics we need one more identity (which follows from the definition of the transition matrix in (3.295) and the detailed balance property) to transform the *tanh* occurring in the derivative of $\tilde{H}$:

$$\begin{aligned} \tanh[\beta h_j(\sigma')] p_{\text{eq}}(\sigma') &= \sum_{\sigma''} \sigma_j'' W[\sigma''; \sigma'] p_{\text{eq}}(\sigma') \\ &= \sum_{\sigma''} W[\sigma'; \sigma''] \sigma_j'' p_{\text{eq}}(\sigma''). \end{aligned}$$

For parallel dynamics ℓ and ℓ' are the real time labels t and t' , and we get, with $\tau = t - t'$:

$$G_{ij}(\tau > 0) = -\beta [C_{ij}(\tau+1) - C_{ij}(\tau-1)], \quad G_{ij}(\tau \leq 0) = 0. \quad (3.326)$$

For the continuous-time version (3.276) of sequential dynamics the time t is defined as $t = \ell/N$, and the difference equation (3.325) becomes a differential

equation. For perturbations at time t' in the definition of the response function (3.323) to retain a non-vanishing effect at (re-scaled) time t in the limit $N \rightarrow \infty$, they will have to be re-scaled as well: $\delta\theta_j(t') \rightarrow N\delta\theta_j(t')$. As a result:

$$G_{ij}(\tau) = -\beta\theta(\tau)\frac{d}{d\tau}C_{ij}(\tau). \quad (3.327)$$

The need to re-scale perturbations in making the transition from discrete to continuous times has the same origin as the need to re-scale the random forces in the derivation of the continuous-time Langevin equation from a discrete-time process. Going from ordinary derivatives to function derivatives (which is what happens in the continuous-time limit), implies replacing Kronecker delta's $\delta_{t,t'}$ by Dirac delta-functions according to $\delta_{t,t'} \rightarrow \Delta\delta(t-t')$, where Δ is the average duration of an iteration step. Equations (3.326) and (3.327) are examples of so-called *fluctuation-dissipation theorems* (FDT).

For systems described by a Fokker-Planck equation (3.221) the simplest way to calculate correlation- and response-functions is by first returning to the underlying discrete-time system and leaving the continuous time limit $\Delta \rightarrow 0$ until the end. We saw above that for small but finite time-steps Δ the underlying discrete-time process is described by [Coo01, SC00, SC01, CKS05]

$$t = \ell\Delta, \quad p_{\ell\Delta+\Delta}(\boldsymbol{\sigma}) = [1 + \Delta\mathcal{L}\boldsymbol{\sigma} + \mathcal{O}(\Delta^{\frac{3}{2}})]p_{\ell\Delta}(\boldsymbol{\sigma}),$$

with $\ell = 0, 1, 2, \dots$ and with the differential operator

$$\mathcal{L}\boldsymbol{\sigma} = -\frac{\partial}{\partial\sigma_i}[f_i(\boldsymbol{\sigma}) - T\frac{\partial}{\partial\sigma_i}].$$

From this it follows that the conditional probability density $p_{\ell\Delta}(\boldsymbol{\sigma}|\boldsymbol{\sigma}', \ell'\Delta)$ for finding state $\boldsymbol{\sigma}$ at time $\ell\Delta$, given the system was in state $\boldsymbol{\sigma}'$ at time $\ell'\Delta$, must be

$$p_{\ell\Delta}(\boldsymbol{\sigma}|\boldsymbol{\sigma}', \ell'\Delta) = [1 + \Delta\mathcal{L}\boldsymbol{\sigma} + \mathcal{O}(\Delta^{\frac{3}{2}})]^{\ell-\ell'}\delta[\boldsymbol{\sigma} - \boldsymbol{\sigma}']. \quad (3.328)$$

Equation (3.328) will be our main building block. Firstly, we will calculate the correlations:

$$\begin{aligned} C_{ij}(\ell\Delta, \ell'\Delta) &= \langle \sigma_i(\ell\Delta)\sigma_j(\ell'\Delta) \rangle \\ &= \int d\boldsymbol{\sigma}d\boldsymbol{\sigma}' \sigma_i\sigma'_j p_{\ell\Delta}(\boldsymbol{\sigma}|\boldsymbol{\sigma}', \ell'\Delta)p_{\ell'\Delta}(\boldsymbol{\sigma}') \\ &= \int d\boldsymbol{\sigma} \sigma_i [1 + \Delta\mathcal{L}\boldsymbol{\sigma} + \mathcal{O}(\Delta^{\frac{3}{2}})]^{\ell-\ell'} \int d\boldsymbol{\sigma}' \sigma'_j \delta[\boldsymbol{\sigma} - \boldsymbol{\sigma}'] p_{\ell'\Delta}(\boldsymbol{\sigma}') \\ &= \int d\boldsymbol{\sigma} \sigma_i [1 + \Delta\mathcal{L}\boldsymbol{\sigma} + \mathcal{O}(\Delta^{\frac{3}{2}})]^{\ell-\ell'} [\sigma_j p_{\ell'\Delta}(\boldsymbol{\sigma})]. \end{aligned}$$

At this stage, we can take the limits $\Delta \rightarrow 0$ and $\ell, \ell' \rightarrow \infty$, with $t = \ell\Delta$ and $t' = \ell'\Delta$ finite, using $\lim_{\Delta \rightarrow 0} [1 + \Delta A]^{k/\Delta} = e^{kA}$:

$$C_{ij}(t, t') = \int d\boldsymbol{\sigma} \sigma_i e^{(t-t')\mathcal{L}_\sigma} [\sigma_j p_{t'}(\boldsymbol{\sigma})]. \quad (3.329)$$

Next we turn to the response function. A perturbation applied at time $t' = \ell' \Delta$ to the Langevin forces $f_i(\boldsymbol{\sigma})$ comes in at the transition $\boldsymbol{\sigma}(\ell' \Delta) \rightarrow \boldsymbol{\sigma}(\ell' \Delta + \Delta)$. As with sequential dynamics binary networks, the perturbation is re-scaled with the step size Δ to retain significance as $\Delta \rightarrow 0$:

$$\begin{aligned} G_{ij}(\ell \Delta, \ell' \Delta) &= \frac{\partial \langle \sigma_i(\ell \Delta) \rangle}{\Delta \partial \theta_j(\ell' \Delta)} = \frac{\partial}{\Delta \partial \theta_j(\ell' \Delta)} \int d\boldsymbol{\sigma} d\boldsymbol{\sigma}' \sigma_i p_{\ell \Delta}(\boldsymbol{\sigma} | \boldsymbol{\sigma}', \ell' \Delta) p_{\ell' \Delta}(\boldsymbol{\sigma}') \\ &= \int d\boldsymbol{\sigma} d\boldsymbol{\sigma}' d\boldsymbol{\sigma}'' \sigma_i p_{\ell \Delta}(\boldsymbol{\sigma} | \boldsymbol{\sigma}'', \ell' \Delta + \Delta) \left[\frac{\partial p_{\ell' \Delta + \Delta}(\boldsymbol{\sigma}' | \boldsymbol{\sigma}', \ell' \Delta)}{\Delta \partial \theta_j} \right] p_{\ell' \Delta}(\boldsymbol{\sigma}') \\ &= \int d\boldsymbol{\sigma} d\boldsymbol{\sigma}' d\boldsymbol{\sigma}'' \sigma_i [1 + \Delta \mathcal{L}_\sigma + \mathcal{O}(\Delta^{\frac{3}{2}})]^{\ell - \ell' - 1} \delta[\boldsymbol{\sigma} - \boldsymbol{\sigma}''] \times \\ &\quad \left[\frac{1}{\Delta} \frac{\partial}{\partial \theta_j} [1 + \Delta \mathcal{L}_{\boldsymbol{\sigma}''} + \mathcal{O}(\Delta^{\frac{3}{2}})] \delta[\boldsymbol{\sigma}'' - \boldsymbol{\sigma}'] \right] p_{\ell' \Delta}(\boldsymbol{\sigma}') \\ &= - \int d\boldsymbol{\sigma} d\boldsymbol{\sigma}' d\boldsymbol{\sigma}'' \sigma_i [1 + \Delta \mathcal{L}_\sigma + \mathcal{O}(\Delta^{\frac{3}{2}})]^{\ell - \ell' - 1} \times \\ &\quad \delta[\boldsymbol{\sigma} - \boldsymbol{\sigma}''] \delta[\boldsymbol{\sigma}'' - \boldsymbol{\sigma}'] \left[\frac{\partial}{\partial \sigma_j} + \mathcal{O}(\Delta^{\frac{1}{2}}) \right] p_{\ell' \Delta}(\boldsymbol{\sigma}') \\ &= - \int d\boldsymbol{\sigma} \sigma_i [1 + \Delta \mathcal{L}_\sigma + \mathcal{O}(\Delta^{\frac{3}{2}})]^{\ell - \ell' - 1} \left[\frac{\partial}{\partial \sigma_j} + \mathcal{O}(\Delta^{\frac{1}{2}}) \right] p_{\ell' \Delta}(\boldsymbol{\sigma}). \end{aligned}$$

We take the limits $\Delta \rightarrow 0$ and $\ell, \ell' \rightarrow \infty$, with $t = \ell \Delta$ and $t' = \ell' \Delta$ finite:

$$G_{ij}(t, t') = - \int d\boldsymbol{\sigma} \sigma_i e^{(t-t')\mathcal{L}_\sigma} \frac{\partial}{\partial \sigma_j} p_{t'}(\boldsymbol{\sigma}). \quad (3.330)$$

Equations (3.329) and (3.330) apply to arbitrary systems described by Fokker-Planck equations. In the case of conservative forces, i.e., $f_i(\boldsymbol{\sigma}) = -\partial H(\boldsymbol{\sigma})/\partial \sigma_i$, and when the system is in an equilibrium state at time t' so that $C_{ij}(t, t') = C_{ij}(t - t')$ and $G_{ij}(t, t') = G_{ij}(t - t')$, we can take a further step using $p_{t'}(\boldsymbol{\sigma}) = p_{\text{eq}}(\boldsymbol{\sigma}) = Z^{-1} e^{-\beta H(\boldsymbol{\sigma})}$. In that case, taking the time derivative of expression (3.329) gives

$$\frac{\partial}{\partial \tau} C_{ij}(\tau) = \int d\boldsymbol{\sigma} \sigma_i e^{\tau \mathcal{L}_\sigma} \mathcal{L}_\sigma [\sigma_j p_{\text{eq}}(\boldsymbol{\sigma})].$$

Working out the key term in this expression gives

$$\begin{aligned} \mathcal{L}_\sigma [\sigma_j p_{\text{eq}}(\boldsymbol{\sigma})] &= - \sum_i \frac{\partial}{\partial \sigma_i} [f_i(\boldsymbol{\sigma}) - T \frac{\partial}{\partial \sigma_i}] [\sigma_j p_{\text{eq}}(\boldsymbol{\sigma})] \\ &= T \frac{\partial}{\partial \sigma_j} p_{\text{eq}}(\boldsymbol{\sigma}) - \sum_i \frac{\partial}{\partial \sigma_i} [\sigma_j J_i(\boldsymbol{\sigma})], \end{aligned}$$

with the components of the probability current density $J_i(\boldsymbol{\sigma}) = [f_i(\boldsymbol{\sigma}) - T \frac{\partial}{\partial \sigma_i}] p_{\text{eq}}(\boldsymbol{\sigma})$. In equilibrium, however, the current is zero by definition, so only the first term in the above expression survives. Insertion into our previous equation for $\partial C_{ij}(\tau)/\partial \tau$, and comparison with (3.330) leads to the FDT for continuous systems [Coo01, SC00, SC01, CKS05]:

$$\text{Continuous Dynamics:} \quad G_{ij}(\tau) = -\beta \theta(\tau) \frac{d}{d\tau} C_{ij}(\tau).$$

We will now calculate the correlation and response functions explicitly, and verify the validity or otherwise of the FDT relations, for attractor networks away from saturation.

Simple Attractor Networks with Binary Neurons

We will consider the continuous time version (3.276) of the sequential dynamics, with the local fields $h_i(\boldsymbol{\sigma}) = J_{ij}\sigma_j + \theta_i$, and the separable interaction matrix (3.292). We already solved the dynamics of this model for the case with zero external fields and away from saturation (i.e., $p \ll \sqrt{N}$). Having non-zero, or even time-dependent, external fields does not affect the calculation much; one adds the external fields to the internal ones and finds the macroscopic laws (3.293) for the overlaps with the stored patterns being replaced by [Coo01, SC00, SC01, CKS05]

$$\dot{\mathbf{m}}(t) = \lim_{N \rightarrow \infty} \frac{1}{N} \boldsymbol{\xi}_i \tanh [\beta \boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m}(t) + \theta_i(t)] - \mathbf{m}(t), \quad (3.331)$$

Fluctuations in the local fields are of vanishing order in N (since the fluctuations in $\mathbf{m}$ are), so that one can easily derive from the master equation (3.276) the following expressions for spin averages:

$$\frac{d}{dt} \langle \sigma_i(t) \rangle = \tanh \beta [\boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m}(t) + \theta_i(t)] - \langle \sigma_i(t) \rangle, \quad (3.332)$$

$$\begin{aligned} i \neq j: \quad \frac{d}{dt} \langle \sigma_i(t) \sigma_j(t) \rangle &= \tanh \beta [\boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m}(t) + \theta_i(t)] \langle \sigma_j(t) \rangle \\ &+ \tanh \beta [\boldsymbol{\xi}_j \cdot \mathbf{A} \mathbf{m}(t) + \theta_j(t)] \langle \sigma_i(t) \rangle - 2 \langle \sigma_i(t) \sigma_j(t) \rangle. \end{aligned}$$

Correlations at different times are calculated by applying (3.332) to situations where the microscopic state at time t' is known exactly, i.e., where $p_{t'}(\boldsymbol{\sigma}) = \delta_{\boldsymbol{\sigma}, \boldsymbol{\sigma}'}$ for some $\boldsymbol{\sigma}'$:

$$\begin{aligned} \langle \sigma_i(t) \rangle |_{\boldsymbol{\sigma}(t')=\boldsymbol{\sigma}'} &= \sigma'_i e^{-(t-t')} + \int_{t'}^t ds e^{s-t} \tanh \beta \times \\ &\times [\boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m}(s; \boldsymbol{\sigma}', t') + \theta_i(s)], \end{aligned} \quad (3.333)$$

with $\mathbf{m}(s; \boldsymbol{\sigma}', t')$ denoting the solution of (3.331) following initial condition $\mathbf{m}(t') = \frac{1}{N} \sigma'_i \boldsymbol{\xi}_i$. If we multiply both sides of (3.333) by σ'_j and average over all possible states $\boldsymbol{\sigma}'$ at time t' we get in leading order in N :

$$\langle \sigma_i(t) \sigma_j(t') \rangle = \langle \sigma_i(t') \sigma_j(t') \rangle e^{-(t-t')} + \int_{t'}^t ds e^{s-t} \langle \tanh \beta [\boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m}(s; \boldsymbol{\sigma}(t'), t') + \theta_i(s)] \sigma_j(t') \rangle.$$

Because of the existence of deterministic laws for the overlaps $\mathbf{m}$ in the $N \rightarrow \infty$ limit, we know with probability one that during the stochastic process the actual value $\mathbf{m}(\boldsymbol{\sigma}(t'))$ must be given by the solution of (3.331), evaluated at time t' . As a result we get, with $C_{ij}(t, t') = \langle \sigma_i(t) \sigma_j(t') \rangle$:

$$C_{ij}(t, t') = C_{ij}(t', t') e^{-(t-t')} + \int_{t'}^t ds e^{s-t} \tanh \beta [\boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m}(s) + \theta_i(s)] \langle \sigma_j(t') \rangle. \quad (3.334)$$

Similarly we get from the solution of (3.332) an equation for the leading order in N of the response functions, by derivation with respect to external fields:

$$\begin{aligned} \frac{\partial \langle \sigma_i(t) \rangle}{\partial \theta_j(t')} &= \beta \theta(t-t') \int_{-\infty}^t ds e^{s-t} [1 - \tanh^2 \beta [\boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m}(s) + \theta_i(s)]] \times \\ &\times \left[\frac{1}{N} (\boldsymbol{\xi}_i \cdot \mathbf{A} \boldsymbol{\xi}_k) \frac{\partial \langle \sigma_k(s) \rangle}{\partial \theta_j(t')} + \delta_{ij} \delta(s-t') \right], \quad \text{or} \end{aligned}$$

$$\begin{aligned} G_{ij}(t, t') &= \beta \delta_{ij} \theta(t-t') e^{-(t-t')} [1 - \tanh^2 \beta [\boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m}(t') + \theta_i(t')]] \quad (3.335) \\ &+ \beta \theta(t-t') \int_{t'}^t ds e^{s-t} \times \\ &\times [1 - \tanh^2 \beta [\boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m}(s) + \theta_i(s)]] \frac{1}{N} (\boldsymbol{\xi}_i \cdot \mathbf{A} \boldsymbol{\xi}_k) G_{kj}(s, t'). \end{aligned}$$

For $t = t'$ we retain in leading order in N only the instantaneous single-site contribution

$$\lim_{t' \uparrow t} G_{ij}(t, t') = \beta \delta_{ij} [1 - \tanh^2 \beta [\boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m}(t) + \theta_i(t)]] . \quad (3.336)$$

This leads to the following ansatz for the scaling with N of the $G_{ij}(t, t')$, which can be shown to be correct by insertion into (3.335), in combination with the correctness at $t = t'$ following from (3.336):

$$i = j : \quad G_{ii}(t, t') = \mathcal{O}(1), \quad i \neq j : \quad G_{ij}(t, t') = \mathcal{O}(N^{-1})$$

Note that this implies $\frac{1}{N} (\boldsymbol{\xi}_i \cdot \mathbf{A} \boldsymbol{\xi}_k) G_{kj}(s, t') = \mathcal{O}(\frac{1}{N})$. In leading order in N we now find

$$G_{ij}(t, t') = \beta \delta_{ij} \theta(t - t') e^{-(t-t')} [1 - \tanh^2 \beta [\boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m}(t') + \theta_i(t')]] . \quad (3.337)$$

For those cases where the macroscopic laws (3.331) describe evolution to a stationary state $\mathbf{m}$, obviously requiring stationary external fields $\theta_i(t) = \theta_i$, we can take the limit $t \rightarrow \infty$, with $t - t' = \tau$ fixed, in (3.334) and (3.337). Using the $t \rightarrow \infty$ limits of (3.332) we subsequently find time translation invariant expressions: $\lim_{t \rightarrow \infty} C_{ij}(t, t - \tau) = C_{ij}(\tau)$ and $\lim_{t \rightarrow \infty} G_{ij}(t, t - \tau) = G_{ij}(\tau)$, with in leading order in N

$$C_{ij}(\tau) = \tanh \beta [\boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m} + \theta_i] \tanh \beta [\boldsymbol{\xi}_j \cdot \mathbf{A} \mathbf{m} + \theta_j] \\ + \delta_{ij} e^{-\tau} [1 - \tanh^2 \beta [\boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m} + \theta_i]] ,$$

$$G_{ij}(\tau) = \beta \delta_{ij} \theta(\tau) e^{-\tau} [1 - \tanh^2 \beta [\boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m} + \theta_i]] ,$$

for which the fluctuation–dissipation theorem (3.327) holds [Coo01, SC00, SC01, CKS05]:

$$G_{ij}(\tau) = -\beta \theta(\tau) \frac{d}{d\tau} C_{ij}(\tau) .$$

We now turn to the parallel dynamical rules (3.295), with the local fields $h_i(\boldsymbol{\sigma}) = J_{ij} \sigma_j + \theta_i$, and the interaction matrix (3.292). As before, having time–dependent external fields amounts simply to adding these fields to the internal ones, and the dynamic laws (3.302) are found to be replaced by

$$\mathbf{m}(t+1) = \lim_{N \rightarrow \infty} \frac{1}{N} \boldsymbol{\xi}_i \tanh [\beta \boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m}(t) + \theta_i(t)] . \quad (3.338)$$

Fluctuations in the local fields are again of vanishing order in N , and the parallel dynamics versions of equations (3.332), to be derived from (3.295), are found to be

$$\langle \sigma_i(t+1) \rangle = \tanh \beta [\boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m}(t) + \theta_i(t)] , \quad (3.339)$$

$$i \neq j : \quad \langle \sigma_i(t+1) \sigma_j(t+1) \rangle = \\ \tanh \beta [\boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m}(t) + \theta_i(t)] \tanh \beta [\boldsymbol{\xi}_j \cdot \mathbf{A} \mathbf{m}(t) + \theta_j(t)] . \quad (3.340)$$

With $\mathbf{m}(t; \boldsymbol{\sigma}', t')$ denoting the solution of the map (3.338) following initial condition $\mathbf{m}(t') = \frac{1}{N} \boldsymbol{\sigma}'_i \boldsymbol{\xi}_i$, we immediately get from equations (3.339, 3.340) the correlation functions:

$$C_{ij}(t, t) = \delta_{ij} + [1 - \delta_{ij}] \tanh \beta [\boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m}(t-1) + \theta_i(t-1)] \times \\ \tanh \beta [\boldsymbol{\xi}_j \cdot \mathbf{A} \mathbf{m}(t-1) + \theta_j(t-1)] ,$$

$$t > t' : \quad C_{ij}(t, t') = \langle \tanh \beta [\boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m}(t-1; \boldsymbol{\sigma}(t'), t') + \theta_i(t-1)] \sigma_j(t') \rangle \\ = \tanh \beta [\boldsymbol{\xi}_i \cdot \mathbf{A} \mathbf{m}(t-1) + \theta_i(t-1)] \tanh \beta [\boldsymbol{\xi}_j \cdot \mathbf{A} \mathbf{m}(t'-1) + \theta_j(t'-1)] .$$

From (3.339) also follow equations determining the leading order in N of the response functions $G_{ij}(t, t')$, by derivation with respect to the external fields $\theta_j(t')$:

$$\begin{aligned} t' > t - 1 : G_{ij}(t, t') &= 0, \\ t' = t - 1 : G_{ij}(t, t') &= \beta \delta_{ij} [1 - \tanh^2 \beta [\boldsymbol{\xi}_i \cdot \mathbf{A}\mathbf{m}(t - 1) + \theta_i(t - 1)]] , \\ t' < t - 1 : G_{ij}(t, t') &= \beta [1 - \tanh^2 \beta [\boldsymbol{\xi}_i \cdot \mathbf{A}\mathbf{m}(t - 1) + \theta_i(t - 1)]] \times \\ &\times \frac{1}{N} (\boldsymbol{\xi}_i \cdot \mathbf{A} \boldsymbol{\xi}_k) G_{kj}(t - 1, t'). \end{aligned}$$

It now follows iteratively that all off-diagonal elements must be of vanishing order in N : $G_{ij}(t, t - 1) = \delta_{ij} G_{ii}(t, t - 1) \rightarrow G_{ij}(t, t - 2) = \delta_{ij} G_{ii}(t, t - 2) \rightarrow \dots$, so that in leading order

$$G_{ij}(t, t') = \beta \delta_{ij} \delta_{t, t' + 1} [1 - \tanh^2 \beta [\boldsymbol{\xi}_i \cdot \mathbf{A}\mathbf{m}(t') + \theta_i(t')]] .$$

For those cases where the macroscopic laws (3.338) describe evolution to a stationary state $\mathbf{m}$, with stationary external fields, we can take the limit $t \rightarrow \infty$, with $t - t' = \tau$ fixed above. We find time translation invariant expressions: $\lim_{t \rightarrow \infty} C_{ij}(t, t - \tau) = C_{ij}(\tau)$ and $\lim_{t \rightarrow \infty} G_{ij}(t, t - \tau) = G_{ij}(\tau)$, with in leading order in N :

$$\begin{aligned} C_{ij}(\tau) &= \tanh \beta [\boldsymbol{\xi}_i \cdot \mathbf{A}\mathbf{m} + \theta_i] \tanh \beta [\boldsymbol{\xi}_j \cdot \mathbf{A}\mathbf{m} + \theta_j] \\ &\quad + \delta_{ij} \delta_{\tau, 0} [1 - \tanh^2 \beta [\boldsymbol{\xi}_i \cdot \mathbf{A}\mathbf{m} + \theta_i]] , \\ G_{ij}(\tau) &= \beta \delta_{ij} \delta_{\tau, 1} [1 - \tanh^2 \beta [\boldsymbol{\xi}_i \cdot \mathbf{A}\mathbf{m} + \theta_i]] , \end{aligned}$$

obeying the Fluctuation-Dissipation Theorem (3.326):

$$G_{ij}(\tau > 0) = -\beta [C_{ij}(\tau + 1) - C_{ij}(\tau - 1)] .$$

Graded-Response Neurons with Uniform Synapses

Let us finally find out how to calculate correlation and response function for the simple network (3.222) of graded-response neurons, with (possibly time-dependent) external forces $\theta_i(t)$, and with uniform synapses $J_{ij} = J/N$ [Coo01, SC00, SC01, CKS05]:

$$\dot{u}_i(t) = \frac{J}{N} \sum_j g[\gamma u_j(t)] - u_i(t) + \theta_i(t) + \eta_i(t). \quad (3.341)$$

For a given realization of the external forces and the Gaussian noise variables $\{\eta_i(t)\}$ we can formally integrate (3.341) and find

$$u_i(t) = u_i(0) e^{-t} + \int_0^t ds e^{s-t} \left[J \int du \rho(u; \mathbf{u}(s)) g[\gamma u] + \theta_i(s) + \eta_i(s) \right], \quad (3.342)$$

with the distribution of membrane potentials $\rho(u; \mathbf{u}) = N^{-1} \sum_i \delta[u - u_i]$. The correlation function $C_{ij}(t, t') = \langle u_i(t) u_j(t') \rangle$ immediately follows from (3.342). Without loss of generality we can define $t \geq t'$. For absent external forces (which were only needed in order to define the response function), and upon using $\langle \eta_i(s) \rangle = 0$ and $\langle \eta_i(s) \eta_j(s') \rangle = 2T \delta_{ij} \delta(s - s')$, we arrive at

$$\begin{aligned} C_{ij}(t, t') &= T \delta_{ij} (e^{t'-t} - e^{-t'-t}) + \\ &\langle \left[u_i(0) e^{-t} + J \int du g[\gamma u] \int_0^t ds e^{s-t} \rho(u; \mathbf{u}(s)) \right] \times \\ &\times \left[u_j(0) e^{-t'} + J \int du g[\gamma u] \int_0^{t'} ds' e^{s'-t'} \rho(u; \mathbf{u}(s')) \right] \rangle. \end{aligned}$$

For $N \rightarrow \infty$, however, we know the distribution of potentials to evolve deterministically: $\rho(u; \mathbf{u}(s)) \rightarrow \rho_s(u)$ where $\rho_s(u)$ is the solution of (3.309). This allows us to simplify the above expression to

$$\begin{aligned} N \rightarrow \infty : \quad C_{ij}(t, t') &= T \delta_{ij} (e^{t'-t} - e^{-t'-t}) \quad (3.343) \\ &+ \langle \left[u_i(0) e^{-t} + J \int du g[\gamma u] \int_0^t ds e^{s-t} \rho_s(u) \right] \times \\ &\times \left[u_j(0) e^{-t'} + J \int du g[\gamma u] \int_0^{t'} ds' e^{s'-t'} \rho_{s'}(u) \right] \rangle. \end{aligned}$$

Next we turn to the response function $G_{ij}(t, t') = \delta \langle u_i(t) \rangle / \delta \xi_j(t')$ (its definition involves functional rather than scalar differentiation, since time is continuous). After this differentiation the forces $\{\theta_i(s)\}$ can be put to zero. Functional differentiation of (3.342), followed by averaging, then leads us to

$$\begin{aligned} G_{ij}(t, t') &= \theta(t - t') \delta_{ij} e^{t'-t} - J \int du g[\gamma u] \frac{\partial}{\partial u} \times \\ &\int_0^t ds e^{s-t} \frac{1}{N} \sum_k \lim_{\boldsymbol{\theta} \rightarrow \mathbf{0}} \langle \delta[u - u_k(s)] \frac{\delta u_k(s)}{\delta \theta_j(t')} \rangle. \end{aligned}$$

In view of (3.342) we make the self-consistent ansatz $\delta u_k(s) / \delta \xi_j(s') = \mathcal{O}(N^{-1})$ for $k \neq j$. This produces

$$N \rightarrow \infty : \quad G_{ij}(t, t') = \theta(t - t') \delta_{ij} e^{t'-t}.$$

Since equation (3.309) evolves towards a stationary state, we can also take the limit $t \rightarrow \infty$, with $t - t' = \tau$ fixed, in (3.343). Assuming non-pathological decay of the distribution of potentials allows us to put $\lim_{t \rightarrow \infty} \int_0^t ds e^{s-t} \rho_s(u) = \rho(u)$ (the stationary solution of (3.309)), with which we find also (3.343) reducing to time translation invariant expressions for $N \rightarrow \infty$, $\lim_{t \rightarrow \infty} C_{ij}(t, t - \tau) = C_{ij}(\tau)$ and $\lim_{t \rightarrow \infty} G_{ij}(t, t - \tau) = G_{ij}(\tau)$, in which

$$C_{ij}(\tau) = T\delta_{ij}e^{-\tau} + J^2 \left\{ \int du \rho(u)g[\gamma u] \right\}^2, \\ G_{ij}(\tau) = \theta(\tau)\delta_{ij}e^{-\tau}.$$

Clearly the leading orders in N of these two functions obey the fluctuation-dissipation theorem:

$$G_{ij}(\tau) = -\beta\theta(\tau)\frac{d}{d\tau}C_{ij}(\tau).$$

As with the binary neuron attractor networks for which we calculated the correlation and response functions earlier, the impact of detailed balance violation (occurring when $A_{\mu\nu} \neq A_{\nu\mu}$ in networks with binary neurons and synapses (3.292), and in all networks with graded-response neurons on the validity of the fluctuation-dissipation theorems, vanishes for $N \rightarrow \infty$, provided our networks are relatively simple and evolve to a stationary state in terms of the macroscopic observables (the latter need not necessarily happen. Detailed balance violation, however, would be noticed in the finite size effects [CCV98].

3.4 Synchronization in Arrays of Josephson Junctions

The *synchronization of coupled nonlinear oscillators* has been a fertile area of research for decades [PRK01]. In particular, *Winfree-type phase models* [Win67] have been extensively studied. In 1D, a generic version of this model for N oscillators reads

$$\dot{\theta}_j = \Omega_j + \sum_{k=1}^N \sigma_{j,k} \Gamma(\theta_k - \theta_j), \quad (3.344)$$

where θ_j is the phase of oscillator j , which can be envisioned as a point moving around the unit circle with angular velocity $\dot{\theta}_j = d\theta_j/dt$. In the absence of coupling, this overdamped oscillator has an angular velocity Ω_j . $\Gamma(\theta_k - \theta_j)$ is the coupling function, and $\sigma_{j,k}$ describes the range and nature (e.g., attractive or repulsive) of the coupling. The special case

$$\Gamma(\theta_k - \theta_j) = \sin(\theta_k - \theta_j), \quad \sigma_{j,k} = \alpha/N, \quad \alpha = \text{const},$$

corresponds to the uniform, sinusoidal coupling of each oscillator to the remaining $N-1$ oscillators. This mean-field system is usually called the *globally-coupled Kuramoto model* (GKM). Kuramoto was the first to show that for this particular form of coupling and in the $N \rightarrow \infty$ limit, there is a continuous dynamical phase transition at a critical value of the coupling strength α_c and that for $\alpha > \alpha_c$ both phase and frequency synchronization appear in the system [Kur84, Str00]. If $\sigma_{j,k} = \alpha\delta_{j,k\pm 1}$ while the coupling function retains the form $\Gamma(\theta_j - \theta_k) = \sin(\theta_k - \theta_j)$, then we have the so-called *locally-coupled*

Kuramoto model (LKM), in which each oscillator is coupled only to its nearest neighbors. Studies of synchronization in the LKM [SSK87], including extensions to more than one spatial dimension, have shown that α_c grows without bound in the $N \rightarrow \infty$ limit [Sm88].

Watts and Strogatz introduced a simple model for tuning collections of coupled dynamical systems between the two extremes of random and regular networks [WS98]. In this model, connections between nodes in a regular array are randomly rewired with a probability p , such that $p = 0$ means the network is regularly connected, while $p = 1$ results in a random connection of nodes. For a range of intermediate values of p between these two extremes, the network retains a property of regular networks (a large clustering coefficient) and also acquires a property of random networks (a short characteristic path length between nodes). Networks in this intermediate configuration are termed *small-world networks*. Many examples of such small worlds, both natural and human-made, have been discussed [Str01]. Not surprisingly, there has been much interest in the synchronization of dynamical systems connected in a small-world geometry [BP02, NML03]. Generically, such studies have shown that the presence of small-world connections make it easier for a network to synchronize, an effect generally attributed to the reduced path length between the linked systems. This has also been found to be true for the special case in which the dynamics of each oscillator is described by a Kuramoto model [HCK02a, HCK02b].

As an example of *physically-controllable systems of nonlinear oscillators*, which can be studied both theoretically and experimentally, Josephson junction (JJ) arrays are almost without peer. Through modern fabrication techniques and careful experimental methods one can attain a high degree of control over the dynamics of a JJ array, and many detailed aspects of array behavior have been studied [NLG00]. Among the many different geometries of JJ arrays, *ladder* arrays deserve special attention. For example, they have been observed to support stable time-dependent, spatially-localized states known as discrete breathers [TMO00]. In addition, the ladder geometry is more complex than that of better understood serial arrays but less so than fully two-dimensional (2D) arrays. In fact, a ladder can be considered as a special kind of 2D array, and so the study of ladders could throw some light on the behavior of such 2D arrays. Also, linearly-stable synchronization of the horizontal, or rung, junctions in a ladder is observed in the absence of a load over a wide range of dc bias currents and junction parameters (such as junction capacitance), so that synchronization in this geometry appears to be robust [TSS05].

In the mid 1990's it was shown that a serial array of zero-capacitance, i.e., overdamped, junctions coupled to a load could be mapped onto the GKM [WCS96, WCS98]. The load in this case was essential in providing an all-to-all coupling among the junctions. The result was based on an averaging process, in which (at least) two distinct time scales were identified: the 'short' time scale set by the rapid voltage oscillations of the junctions (the array

was current biased above its critical current) and ‘long’ time scale over which the junctions synchronize their voltages. If the *resistively-shunted junction* (RSJ) equations describing the dynamics of the junctions are integrated over one cycle of the ‘short’ time scale, what remains is the ‘slow’ dynamics, describing the synchronization of the array. This mapping is useful because it allows knowledge about the GKM to be applied to understanding the dynamics of the serial JJ array. For example, the authors of [WCS96] were able, based on the GKM, to predict the level of critical current disorder the array could tolerate before frequency synchronization would be lost. Frequency synchronization, also described as entrainment, refers to the state of the array in which all junctions not in the zero-voltage state have equal (to within some numerical precision) time-averaged voltages: $(\hbar/2e)\langle\dot{\theta}_j\rangle_t$, where θ_j is the gauge-invariant phase difference across junction j . More recently, the ‘slow’ synchronization dynamics of finite-capacitance serial arrays of JJ’s has also been studied [CS95, WS97]. Perhaps surprisingly, however, no experimental work on JJ arrays has verified the accuracy of this GKM mapping. Instead, the first detailed experimental verification of Kuramoto’s theory was recently performed on systems of coupled electrochemical oscillators [KZH02].

Recently, [DDT03] showed, with an eye toward a better understanding of synchronization in 2D JJ arrays, that a ladder array of *overdamped* junctions could be mapped onto the LKM. This work was based on an averaging process, as in [WCS96], and was valid in the limits of weak critical current disorder (less than about 10%) and large dc bias currents, I_B , along the rung junctions ($I_B/\langle I_c \rangle \gtrsim 3$, where $\langle I_c \rangle$ is the arithmetic average of the critical currents of the rung junctions). The result demonstrated, for both open and periodic boundary conditions, that synchronization of the current-biased rung junctions in the ladder is well described by (3.344).

In this section, following [TSS05], we demonstrate that a ladder array of *underdamped junctions* can be mapped onto a second-order Winfree-type oscillator model of the form

$$a\ddot{\theta}_j + \dot{\theta}_j = \Omega_j + \sum_{k=1}^N \sigma_{j,k} \Gamma(\theta_k - \theta_j), \quad (3.345)$$

where a is a constant related to the average capacitance of the rung junctions. This result is based on the resistively and *capacitively-shunted junction* (RCSJ) model and a multiple time scale analysis of the classical equations for the array. Secondly, we study the effects of *small world* (SW) connections on the synchronization of both overdamped and underdamped ladder arrays. It appears that SW connections make it easier for the ladder to synchronize, and that a Kuramoto or Winfree type model (3.344) and (3.345), suitably generalized to include the new connections, accurately describes the synchronization of this ladder.

3.4.1 Phase Model for Underdamped JJJ

Following [TSS05] we analyze synchronization in disordered Josephson junction arrays. The ladder geometry used consists of an array with $N = 8$ plaquettes, periodic boundary conditions, and uniform dc bias currents, I_B , along the rung junctions (see Figure 3.4). The *gauge-invariant phase difference* across rung junction j is γ_j , while the phase difference across the off-rung junctions along the outer(inner) edge of plaquette j is $\psi_{1,j}(\psi_{2,j})$. The critical current, resistance, and capacitance of rung junction j are denoted I_{cj} , R_j , and C_j , respectively. For simplicity, we assume all off-rung junctions are identical, with critical current I_{co} , resistance R_o , and capacitance C_o . We also assume that the product of the junction critical current and resistance is the same for all junctions in the array [Ben95], with a similar assumption about the ratio of each junction's critical current with its capacitance:

$$I_{cj}R_j = I_{co}R_o = \frac{\langle I_c \rangle}{\langle R^{-1} \rangle} \quad (3.346)$$

$$\frac{I_{cj}}{C_j} = \frac{I_{co}}{C_o} = \frac{\langle I_c \rangle}{\langle C \rangle}, \quad (3.347)$$

where for any generic quantity X , the angular brackets with no subscript denote an arithmetic average over the set of rung junctions,

$$\langle X \rangle \equiv (1/N) \sum_{j=1}^N X_j.$$

For convenience, we work with dimensionless quantities. Our dimensionless time variable is

$$\tau \equiv \frac{t}{t_c} = \frac{2e\langle I_c \rangle t}{\hbar \langle R^{-1} \rangle}, \quad (3.348)$$

where t is the ordinary time. In the following, derivatives with respect to τ will be denoted by *prime* (e.g., $\psi' = d\psi/d\tau$). The dimensionless bias current is

$$i_B \equiv \frac{I_B}{\langle I_c \rangle}, \quad (3.349)$$

while the dimensionless critical current of rung junction j is $i_{cj} \equiv I_{cj}/\langle I_c \rangle$. The McCumber parameter in this case is

$$\beta_c \equiv \frac{2e\langle I_c \rangle \langle C \rangle}{\hbar \langle R^{-1} \rangle^2}. \quad (3.350)$$

Note that β_c is proportional to the mean capacitance of the rung junctions. An important dimensionless parameter is

$$\alpha \equiv \frac{I_{co}}{\langle I_c \rangle}, \quad (3.351)$$

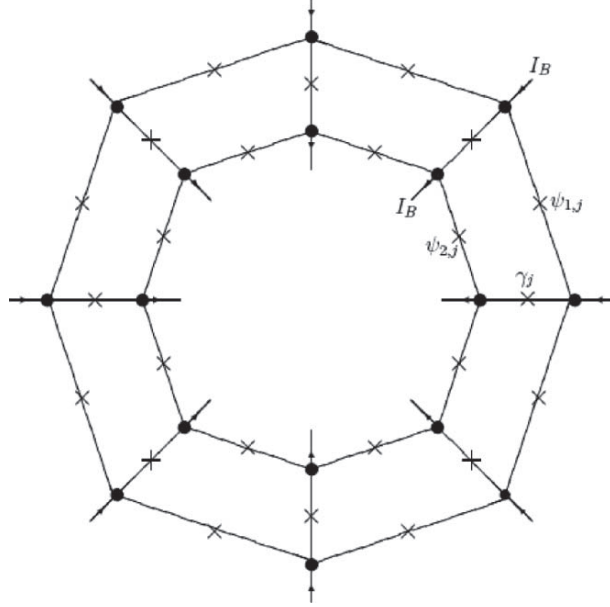


Fig. 3.4. A ladder array of Josephson junctions with periodic boundary conditions and $N = 8$ plaquettes. A uniform, dc bias current I_B is inserted into and extracted from each rung as shown. The gauge-invariant phase difference across the rung junctions is denoted by γ_j where $1 \leq j \leq N$, while the corresponding quantities for the off-rung junctions along the outer(inner) edge are $\psi_{1,j}(\psi_{2,j})$ (adapted and modified from [TSS05]).

which will effectively tune the nearest-neighbor interaction strength in our phase model for the ladder.

Conservation of charge applied to the superconducting islands on the outer and inner edge, respectively, of rung junction j yields the following equations in dimensionless variables [TSS05]:

$$\begin{aligned} i_B - i_{cj} \sin \gamma_j - i_{cj} \gamma'_j - i_{cj} \beta_c \gamma''_j - \alpha \sin \psi_{1,j} - \alpha \psi'_{1,j} \\ - \alpha \beta_c \psi''_{1,j} + \alpha \sin \psi_{1,j-1} + \alpha \psi'_{1,j-1} + \alpha \beta_c \psi''_{1,j-1} = 0, \end{aligned} \quad (3.352)$$

$$\begin{aligned} -i_B + i_{cj} \sin \gamma_j + i_{cj} \gamma'_j + i_{cj} \beta_c \gamma''_j - \alpha \sin \psi_{2,j} - \alpha \psi'_{2,j} \\ - \alpha \beta_c \psi''_{2,j} + \alpha \sin \psi_{2,j-1} + \alpha \psi'_{2,j-1} + \alpha \beta_c \psi''_{2,j-1} = 0, \end{aligned} \quad (3.353)$$

where $1 \leq j \leq N$. The result is a set of $2N$ equations in $3N$ unknowns: γ_j , $\psi_{1,j}$, and $\psi_{2,j}$. We supplement (3.353) by the constraint of fluxoid quantization in the absence of external or induced magnetic flux. For plaquette j this constraint yields the relationship

$$\gamma_j + \psi_{2,j} - \gamma_{j+1} - \psi_{1,j} = 0. \quad (3.354)$$

Equations (3.353) and (3.354) can be solved numerically for the $3N$ phases γ_j , $\psi_{1,j}$ and $\psi_{2,j}$ [TSS05].

We assign the rung junction critical currents in one of two ways, randomly or nonrandomly. We generate random critical currents according to a parabolic *probability distribution function* (PDF) of the form

$$P(i_c) = \frac{3}{4\Delta^3} [\Delta^2 - (i_c - 1)^2], \quad (3.355)$$

where $i_c = I_c/\langle I_c \rangle$ represents a scaled critical current, and Δ determines the spread of the critical currents. Equation (3.355) results in critical currents in the range $1 - \Delta \leq i_c \leq 1 + \Delta$. Note that this choice for the PDF (also used in [WCS96]) avoids extreme critical currents (relative to a mean value of unity) that are occasionally generated by PDF's with tails. The nonrandom method of assigning rung junction critical currents was based on the expression

$$i_{cj} = 1 + \Delta - \frac{2\Delta}{(N-1)^2} [4j^2 - 4(N+1)j + (N+1)^2], \quad 1 \leq j \leq N, \quad (3.356)$$

which results in the i_{cj} values varying quadratically as a function of position along the ladder and falling within the range $1 - \Delta \leq i_{cj} \leq 1 + \Delta$. We usually use $\Delta = 0.05$.

Multiple Time-Scale Analysis

Now, our goal is to derive a Kuramoto-like model for the phase differences across the rung junctions, γ_j , starting with (3.353). We begin with two reasonable assumptions. First, we assume there is a simple phase relationship between the two off-rung junctions in the same plaquette [TSS05]:

$$\psi_{2,j} = -\psi_{1,j}, \quad (3.357)$$

the validity of which has been discussed in detail elsewhere [DDT03, FW95]. As a result, (3.354) reduces to

$$\psi_{1,j} = \frac{\gamma_j - \gamma_{j+1}}{2}, \quad (3.358)$$

which implies that (3.352) can be written as

$$\begin{aligned} i_{cj}\beta_c\gamma_j'' + i_{cj}\gamma_j' + \frac{\alpha\beta_c}{2} [\gamma_{j+1}'' - 2\gamma_j'' + \gamma_{j-1}''] + \frac{\alpha}{2} [\gamma_{j+1}' - 2\gamma_j' + \gamma_{j-1}'] \\ = i_B - i_{cj}\sin\gamma_j + \alpha \sum_{\delta=\pm 1} \sin\left(\frac{\gamma_{j+\delta} - \gamma_j}{2}\right). \end{aligned} \quad (3.359)$$

Our second assumption is that we can neglect the discrete Laplacian terms in (3.359), namely

$$\nabla^2 \gamma'_j \equiv \gamma'_{j+1} - 2\gamma'_j + \gamma'_{j-1} \quad \text{and} \quad \nabla^2 \gamma''_j \equiv \gamma''_{j+1} - 2\gamma''_j + \gamma''_{j-1}.$$

We find numerically, over a wide range of bias currents i_B , *McCumber parameters* β_c , and coupling strengths α that $\nabla^2 \gamma'_j$ and $\nabla^2 \gamma''_j$ oscillate with a time-averaged value of approximately zero. Since the multiple time scale method is similar to averaging over a fast time scale, it seems reasonable to drop these terms. In light of this assumption, (3.359) becomes

$$i_{cj}\beta_c\gamma''_j + i_{cj}\gamma'_j = i_B - i_{cj}\sin\gamma_j + \alpha \sum_{\delta=\pm 1} \sin\left(\frac{\gamma_{j+\delta} - \gamma_j}{2}\right). \quad (3.360)$$

We can use (3.360) as the starting point for a multiple time scale analysis. Following [CS95] and [WS97], we divide (3.360) by i_B and define the following quantities:

$$\tilde{\tau} \equiv i_B \tau, \quad \tilde{\beta}_c \equiv i_B \beta_c, \quad \epsilon = 1/i_B. \quad (3.361)$$

In terms of these scaled quantities, (3.360) can be written as

$$i_{cj}\tilde{\beta}_c \frac{d^2 \gamma_j}{d\tilde{\tau}^2} + i_{cj} \frac{d\gamma_j}{d\tilde{\tau}} + \epsilon i_{cj} \sin\gamma_j - \epsilon \alpha \sum_{\delta} \sin\left(\frac{\gamma_{j+\delta} - \gamma_j}{2}\right) = 1. \quad (3.362)$$

Next, we introduce a series of four (dimensionless) time scales,

$$T_n \equiv \epsilon^n \tilde{\tau}, \quad (n = 0, 1, 2, 3), \quad (3.363)$$

which are assumed to be independent of each other. Note that $0 < \epsilon < 1$ since $\epsilon = 1/i_B$. We can think of each successive time scale, T_n , as being ‘slower’ than the scale before it. For example, T_2 describes a slower time scale than T_1 . The time derivatives in 3.362 can be written in terms of the new time scales, since we can think of $\tilde{\tau}$ as being a function of the four independent T_n ’s, $\tilde{\tau} = \tilde{\tau}(T_0, T_1, T_2, T_3)$. Letting $\partial_n \equiv \partial/\partial T_n$, the first and second time derivatives can be written as [TSS05]

$$\frac{d}{d\tilde{\tau}} = \partial_0 + \epsilon \partial_1 + \epsilon^2 \partial_2 + \epsilon^3 \partial_3 \quad (3.364)$$

$$\frac{d^2}{d\tilde{\tau}^2} = \partial_0^2 + 2\epsilon \partial_0 \partial_1 + \epsilon^2 (2\partial_0 \partial_2 + \partial_1^2) + 2\epsilon^3 (\partial_0 \partial_3 + \partial_1 \partial_2), \quad (3.365)$$

where in (3.365) we have dropped terms of order ϵ^4 and higher.

Next, we expand the phase differences in an ϵ expansion

$$\gamma_j = \sum_{n=0}^{\infty} \epsilon^n \gamma_{n,j}(T_0, T_1, T_2, T_3). \quad (3.366)$$

Substituting this expansion into (3.362) and collecting all terms of order ϵ^0 results in the expression

$$1 = i_{cj} \tilde{\beta}_c \partial_0^2 \gamma_{0,j} + i_{cj} \partial_0 \gamma_{0,j}, \quad (3.367)$$

for which we find the solution

$$\gamma_{0,j} = \frac{T_0}{i_{cj}} + \phi_j(T_1, T_2, T_3), \quad (3.368)$$

where we have ignored a transient term of the form $e^{-T_0/\tilde{\beta}_c}$, and where $\phi_j(T_i)$, ($i = 1, 2, 3$) is assumed constant over the fastest time scale T_0 . Note that the expression for $\gamma_{0,j}$ consists of a rapid phase rotation described by T_0/i_{cj} and slower-scale temporal variations, described by ϕ_j , on top of that overturning. In essence, the goal of this technique is to solve for the dynamical behavior of the slow phase variable, ϕ_j . The resulting differential equation for the ϕ_j is [TSS05]:

$$\begin{aligned} \beta_c \phi_j'' + \phi_j' = \Omega_j + K_j \sum_{\delta=\pm 1} \sin \left[\frac{\phi_{j+\delta} - \phi_j}{2} \right] + L_j \sum_{\delta=\pm 1} \sin \left[3 \left(\frac{\phi_{j+\delta} - \phi_j}{2} \right) \right] \\ + M_j \sum_{\delta=\pm 1} \left\{ \cos \left[\frac{\phi_{j+\delta} - \phi_j}{2} \right] - \cos \left[3 \left(\frac{\phi_{j+\delta} - \phi_j}{2} \right) \right] \right\}, \end{aligned} \quad (3.369)$$

where Ω_j is given by the expression (letting $x_j \equiv i_{cj}/i_B$ for convenience)

$$\Omega_j = \frac{1}{x_j} \left[1 - \frac{x_j^4}{(2\beta_c^2 + x_j^2)} \right], \quad (3.370)$$

and the three coupling strengths are

$$K_j = \frac{\alpha}{i_{cj}} \left[1 + \frac{x_j^4 (3x_j^2 + 23\beta_c^2)}{16 (\beta_c^2 + x_j^2)^2} \right], \quad (3.371)$$

$$L_j = \frac{\alpha}{i_{cj}} \frac{x_j^4 (3\beta_c^2 - x_j^2)}{16 (\beta_c^2 + x_j^2)^2}, \quad (3.372)$$

$$M_j = -\frac{\alpha}{i_{cj}} \frac{x_j^5 \beta_c}{4 (\beta_c^2 + x_j^2)^2}. \quad (3.373)$$

We emphasize that (3.369) is expressed in terms of the original, unscaled, time variable τ and McCumber parameter β_c .

We will generally consider bias current and junction capacitance values such that $x_j^2 \ll \beta_c^2$. In this limit, (3.371)–(3.373) can be approximated as follows [TSS05]:

$$K_j \rightarrow \frac{\alpha}{i_{cj}} \left[1 + \mathcal{O} \left(\frac{1}{i_B^4} \right) \right], \quad (3.374)$$

$$L_j \rightarrow \frac{\alpha}{i_{cj}} \left(\frac{3x_j^4}{16\beta_c^2} \right) \sim \mathcal{O} \left(\frac{1}{i_B^4} \right), \quad (3.375)$$

$$M_j \rightarrow -\frac{\alpha}{i_{cj}} \left(\frac{x_j^5}{4\beta_c^3} \right) \sim \mathcal{O} \left(\frac{1}{i_B^5} \right). \quad (3.376)$$

For large bias currents, it is reasonable to truncate (3.369) at $\mathcal{O}(1/i_B^3)$, which leaves

$$\beta_c \phi_j'' + \phi_j' = \Omega_j + \frac{\alpha}{i_{cj}} \sum_{\delta=\pm 1} \sin \left[\frac{\phi_{j+\delta} - \phi_j}{2} \right], \quad (3.377)$$

where all the cosine coupling terms and the third harmonic sine term have been dropped as a result of the truncation.

In the absence of any coupling between neighboring rung junctions ($\alpha = 0$) the solution to (3.377) is

$$\phi_j^{(\alpha=0)} = A + B e^{-\tau/\beta_c} + \Omega_j \tau,$$

where A and B are arbitrary constants. Ignoring the transient exponential term, we see that $d\phi_j^{(\alpha=0)}/d\tau = \Omega_j$, so we can think of Ω_j as the voltage across rung junction j in the un-coupled limit. Alternatively, Ω_j can be viewed as the angular velocity of the strongly-driven rotator in the un-coupled limit.

Equation (3.377) is our desired phase model for the rung junctions of the underdamped ladder [TSS05]. The result can be described as a locally-coupled Kuramoto model with a second-order time derivative (LKM2) and with junction coupling determined by α . In the context of systems of coupled rotators, the second derivative term is due to the non-negligible rotator inertia, whereas in the case of Josephson junctions the second derivative arises because of the junction capacitance. The *globally-coupled* version of the second-order Kuramoto model (GKM2) has been well studied; in this case the oscillator inertia leads to a first-order synchronization phase transition as well as to hysteresis between a weakly and a strongly coherent synchronized state [TLO97, ABS00].

3.4.2 Comparison of LKM2 and RCSJ Models

We now compare the synchronization behavior of the RCSJ ladder array with the LKM2. We consider frequency and phase synchronization separately. For the rung junctions of the ladder, frequency synchronization occurs when the time average voltages, $\langle v_j \rangle_\tau = \langle \phi_j' \rangle_\tau$ are equal for all N junctions, within some specified precision. In the language of coupled rotators, this corresponds to phase points moving around the unit circle with the same average angular velocity. We quantify the degree of frequency synchronization via an ‘order parameter’ [TSS05]

$$f = 1 - \frac{s_v(\alpha)}{s_v(0)}, \quad (3.378)$$

where $s_v(\alpha)$ is the standard deviation of the N time-average voltages, $\langle v_j \rangle_\tau$:

$$s_v(\alpha) = \sqrt{\frac{\sum_{j=1}^N \left(\langle v_j \rangle_\tau - \frac{1}{N} \sum_{k=1}^N \langle v_k \rangle_\tau \right)^2}{N-1}} \quad (3.379)$$

In general, this standard deviation will be a function of the coupling strength α , so $s_v(0)$ is a measure of the spread of the $\langle v_j \rangle_\tau$ values for N independent junctions. Frequency synchronization of all N junctions is signaled by $f = 1$, while $f = 0$ means all N average voltages have their un-coupled values.

Phase synchronization of the rung junctions is measured by the usual *Kuramoto order parameter*

$$r \equiv \frac{1}{N} \sum_{j=1}^N e^{i\phi_j}. \quad (3.380)$$

Lastly in this subsection, we address the issue of the linear stability of the frequency synchronized states ($\alpha > \alpha_c$) by calculating their *Floquet exponents* numerically for the RCSJ model as well as analytically based on the LKM2, (3.377). The analytic technique used has been described in [TM01], giving as a result for the real part of the Floquet exponents:

$$\text{Re}(\lambda_m t_c) = -\frac{1}{2\beta_c} \left[1 \pm \text{Re} \sqrt{1 - 4\beta_c (\bar{K} + 3\bar{L}) \omega_m^2} \right], \quad (3.381)$$

where stable solutions correspond to exponents, λ_m , with a negative real part. One can think of the ω_m as the normal mode frequencies of the ladder. We find that for a ladder with periodic boundary conditions and N plaquettes

$$\omega_m^2 = \frac{4 \sin^2 \left(\frac{m\pi}{N} \right)}{1 + 2 \sin^2 \left(\frac{m\pi}{N} \right)}, \quad 0 \leq m \leq N-1. \quad (3.382)$$

To arrive at (3.381) we have ignored the effects of disorder so that $\bar{K}$ and $\bar{L}$ are obtained from (3.371) and (3.372) with the substitution $i_{cj} \rightarrow 1$ throughout. This should be reasonable for the levels of disorder we have considered (5%). Substituting the expressions for $\bar{K}$ and $\bar{L}$ into 3.381 results in [TSS05]

$$\text{Re}(\lambda_m t_c) = -\frac{1}{2\beta_c} \left[1 \pm \text{Re} \sqrt{1 - 2\beta_c \alpha \left\{ 1 + \frac{2\beta_c^2}{(i_B^2 \beta_c^2 + 1)^2} \right\} \omega_m^2} \right]. \quad (3.383)$$

We are most interested in the Floquet exponent of minimum magnitude, $\text{Re}(\lambda_{\min} t_c)$, which essentially gives the lifetime of the longest-lived perturbations to the synchronized state.

3.4.3 ‘Small-World’ Connections in JJL Arrays

Many properties of small world networks have been studied in the last several years, including not only the effects of network topology but also the dynamics

of the node elements comprising the network [New00, Str01]. Of particular interest has been the ability of oscillators to synchronize when configured in a small-world manner. Such synchronization studies can be broadly sorted into several categories [TSS05]:

(1) Work on coupled lattice maps has demonstrated that synchronization is made easier by the presence of random, *long-range connections* [GH00, BPV03].

(2) Much attention has been given to the synchronization of continuous time dynamical systems, including the first order *locally-coupled Kuramoto model* (LKM), in the presence of small world connections [HCK02a, HCK02b, Wat99]. For example, Hong and coworkers [HCK02a, HCK02b] have shown that the LKM, which does not exhibit a true dynamical phase transition in the thermodynamic limit ($N \rightarrow \infty$) in the *pristine* case, does exhibit such a phase synchronization transition for even a small number of shortcuts. But the assertion [WC02] that any small world network can synchronize for a given coupling strength and large enough number of nodes, even when the pristine network would not synchronize under the same conditions, is not fully accepted [BP02].

(3) More general studies of synchronization in small world and scale-free networks [BP02, NML03] have shown that the small world topology does not guarantee that a network can synchronize. In [BP02] it was shown that one could calculate the average number of shortcuts per node, s_{sync} , required for a given dynamical system to synchronize. This study found no clear relation between this synchronization threshold and the onset of the small world region, i.e., the value of s such that the average path length between all pairs of nodes in the array is less than some threshold value. [NML03] studied arrays with a power-law distribution of node connectivities (scale-free networks) and found that a broader distribution of connectivities makes a network *less* synchronizable even though the average path length is smaller. It was argued that this behavior was caused by an increased number of connections on the hubs of the scale-free network. Clearly it is dangerous to assume that merely reducing the average path length between nodes of an array will make such an array easier to synchronize.

Now, regarding Josephson-junction arrays, if we have a disordered array biased such that some subset of the junctions are in the voltage state, i.e., undergoing limit cycle oscillations, the question is will the addition of random, long-range connections between junctions aid the array in attaining frequency and/or phase synchronization? Can we address this question by using the mapping discussed above between the RCSJ model for the *underdamped ladder array* and the second-order, locally-coupled Kuramoto model (LKM2). Based on the results of [DDT03], we also know that the RSJ model for an *overdamped ladder* can be mapped onto a first-order, locally-coupled Kuramoto model (LKM). Because of this mapping, the ladder array falls into category (2) of the previous paragraph. In other words, we should expect the existence of

shortcuts to drastically improve the ability of ladder arrays to synchronize [TSS05].

We add connections between pairs of rung junctions that will result in interactions that are longer than nearest neighbor in range. We do so by adding two, nondisordered, off-rung junctions for each such connection. We argue that the RCSJ equations for the underdamped junctions in the ladder array can be mapped onto a straightforward variation of (3.377), in which the sinusoidal coupling term for rung junction j also includes the longer-range couplings due to the added shortcuts. Imagine a ladder with a shortcut between junctions j and l , where $l \neq j, j \pm 1$. Conservation of charge applied to the two superconducting islands that comprise rung junction j will lead to equations very similar to (3.353). For example, the analog to (3.352) will be

$$i_B - i_{cj} \sin \gamma_j - i_{cj} \gamma_j' - \beta_c i_{cj} \gamma_j'' - \alpha \sin \psi_{1,j} - \alpha \psi_{1,j}' - \beta_c \alpha \psi_{1,j}'' + \alpha \sin \psi_{1,j-1} + \alpha \psi_{1,j-1}' + \beta_c \alpha \psi_{1,j-1}'' + \sum_l [\alpha \sin \psi_{1;jl} + \alpha \psi_{1;jl}' + \beta_c \alpha \psi_{1;jl}''] = 0,$$

with an analogous equation corresponding to the inner superconducting island that can be generalized from (3.353). The sum over the index l accounts for all junctions connected to junction j via an added shortcut. Fluxoid quantization still holds, which means that we can augment 3.354 with

$$\gamma_j + \psi_{2;jl} - \gamma_l - \psi_{1;jl} = 0. \quad (3.384)$$

We also assume the analog of (3.357) holds:

$$\psi_{2;jl} = -\psi_{1;jl}. \quad (3.385)$$

Equations (3.384) and (3.385) allow us to write the analog to (3.358) for the case of shortcut junctions:

$$\psi_{1;jl} = \frac{\gamma_j - \gamma_l}{2} \quad (3.386)$$

Equation (3.384), in light of (3.386), can be written as

$$i_B - i_{cj} \sin \gamma_j - i_{cj} \gamma_j' - \beta_c i_{cj} \gamma_j'' + \alpha \sum_{\delta=\pm 1} \sin\left(\frac{\gamma_{j+\delta} - \gamma_j}{2}\right) + \alpha \sum_l \sin\left(\frac{\gamma_j - \gamma_l}{2}\right) + \frac{\alpha}{2} \nabla^2 \gamma_j' + \frac{\alpha}{2} \nabla^2 \gamma_j'' + \frac{\alpha}{2} \sum_l (\gamma_j' - \gamma_l') + \frac{\alpha}{2} \sum_l (\gamma_j'' - \gamma_l'') = 0,$$

where the sums $\sum_l$ are over all rung junctions connected to j via an added shortcut. As we did with the pristine ladder, we will drop the two discrete Laplacians, since they have a very small time average compared to the terms $i_{cj} \gamma_j' + i_{cj} \beta_c \gamma_j''$. The same is also true, however, of the terms $\alpha/2 \sum_l (\gamma_j' - \gamma_l')$ and $\alpha/2 \sum_l (\gamma_j'' - \gamma_l'')$, as direct numerical solution of the full RCSJ equations in the presence of shortcuts demonstrates. So we shall drop these terms as well. Then we have

$$i_B - i_{cj} \sin \gamma_j - i_{cj} \gamma'_j - \beta_c i_{cj} \gamma''_j + \frac{\alpha}{2} \sum_{k \in \Lambda_j} \sin \left(\frac{\gamma_k - \gamma_j}{2} \right), \quad (3.387)$$

where the sum is over all junctions in Λ_j , which is the set of all junctions connected to junction j . From above results we can predict that a multiple time scale analysis of (3.387) results in a phase model of the form

$$\beta_c \frac{d^2 \phi_j}{d\tau^2} + \frac{d\phi_j}{d\tau} = \Omega_j + \frac{\alpha}{2} \sum_{k \in \Lambda_j} \sin \left(\frac{\phi_k - \phi_j}{2} \right), \quad (3.388)$$

where Ω_j is give by (3.370). A similar analysis for the *overdamped ladder* leads to the result

$$\phi'_j = \Omega_j^{(1)} + \frac{\alpha}{2} \sum_{k \in \Lambda_j} \sin \left(\frac{\phi_k - \phi_j}{2} \right), \quad (3.389)$$

where the time-averaged voltage across each overdamped rung junction in the un-coupled limit is

$$\Omega_j^{(1)} = \sqrt{\left(\frac{i_B}{i_{cj}} \right)^2 - 1}. \quad (3.390)$$

Although the addition of shortcuts makes it easier for the array to synchronize, we should also consider the effects of such random connections on the stability of the synchronized state. The Floquet exponents for the synchronized state allow us to quantify this stability. Using a general technique discussed in [PC98], we can calculate the Floquet exponents λ_m for the LKM based on the expression

$$\lambda_m t_c = \alpha E_m^G, \quad (3.391)$$

where E_m^G are the eigenvalues of $\mathbf{G}$, the matrix of coupling coefficients for the array. A specific element, G_{ij} , of this matrix is unity if there is a connection between rung junctions i and j . The diagonal terms, G_{ii} , is merely the negative of the number of junctions connected to junction i . This gives the matrix the property $\sum_j G_{ij} = 0$. In the case of the pristine ladder, the eigenvalues of $\mathbf{G}$ can be calculated analytically, which yields Floquet exponents of the form

$$\lambda_m^{(p=0)} t_c = -4\alpha \sin^2 \left(\frac{m\pi}{N} \right). \quad (3.392)$$

See [TSS05] for more details.

3.5 Global Neurodynamics

3.5.1 A Path-Integral AM-Model

In recent years, theories that can analyze the transient dynamics have been discussed for systems with frustrations especially a *correlation-type associative*

memory [Hop82, SF92, AGS85b, KBH91, MO95, Oka96, MOK96, MOK98a, MOK98b, MKO03, Oka95]. Düring *et al.* presented a *path-integral method* for an *infinite-step sequence processing* model in [DCS98] and analyzed the properties of the stationary state. By using Düring *et al.*'s analysis, Kawamura and Okada succeeded in deriving an exact macroscopic description of the transient dynamics [KO02]. The transient dynamics can be analyzed not only by using the path-integral method [CS93, Coo01] but also by using the *signal-to-noise analysis*, e.g., *statistical neurodynamics* [AM88, Ama88] (see subsection 3.2.9 above). The signal-to-noise analysis is an approximation theory in which crosstalk noise obeys a Gaussian distribution. On the other hand, the path-integral method does not require such a Gaussian approximation of *cross-talk noise*. However surprisingly, the macroscopic equations of the exact solution given by means of the path-integral method are completely equivalent to those of the signal-to-noise analysis with respect to this model.

It has turned out that the infinite-step sequence processing model can be more easily analyzed than the Hopfield model even if it is necessary to treat the dynamical process directly. This reason is as follows. The retrieval state of the infinite-step sequence processing model has no equilibrium state. Therefore, the correlations of the system are not very complex. Since the Hopfield model takes the same states repeatedly, its statistical properties are more complex than the infinite-step sequence processing model. Gardner *et al.* analyzed the transient dynamics of the Hopfield model by using the path-integral method in the case where the dynamics is deterministic [GDM87]. They obtained the macroscopic equations of the transient dynamics at time step t using $\mathcal{O}(t^2)$ macroscopic variables and also obtained the macroscopic equations of the equilibrium state from the transient dynamics. These are equivalent to *replica symmetric* (RS) solutions given by using the replica method [GDM87]. Recently in the Hopfield model, Bolle *et al.* compared the transient dynamics of the path-integral method with those of the signal-to-noise analysis only for a few time steps in the dynamics [BBV04]. They have pointed out that the signal-to-noise analysis is exact up to time step 3 and inexact to step 4 or beyond.

In order to discuss the relation between the path-integral method and the signal-to-noise analysis in more detail, we analyze a *finite-step sequence processing* model, which includes the Hopfield model and the infinite-step sequence processing model in special cases. In the finite-step sequence processing model, the steady states of the system become limit cycles. Since the finite-step sequence processing model can store limit cycles in the dynamics, the properties of the system are periodic and dynamic essentially like the infinite-step sequence processing model. Moreover, the statistical properties of the finite-step sequence processing model are more complex than the infinite-step one. Since the period of the limit cycle is finite, the network takes the same states repeatedly. Namely, the finite-step sequence processing model has the theoretical difficulties of both the Hopfield model and the infinite-step sequence processing model. In this point of view, it would be very interesting

to theoretically discuss the properties of the finite-step sequence processing model.

In this subsection, following [MKO04], we exactly derive the transient dynamics of macroscopic recursive equations with respect to the finite-step sequence processing model by means of the path-integral analysis. Until now, only in the infinite-step sequence processing model, which has no self-interactions, Düring *et al.* derived the stationary state equations of the order parameters by using the path-integral analysis [DCS98]. The transient dynamics of various disordered systems can be also analyzed by using the path-integral method. Therefore, it is important to derive stationary state equations of the order parameters from the macroscopic recursive equations. Assuming the Maxwell construction ansatz, we derive the stationary state equations from the macroscopic recursive equations with respect to the model, which has self-interactions, i.e., the finite-step sequence processing model.

We also analyze the finite-step sequence processing model by means of the signal-to-noise analysis (SCSNA). The stationary state equations given by the path-integral analysis are equivalent to those of the signal-to-noise analysis. This result corresponds to the fact that the replica method and the signal-to-noise analysis give completely equivalent results in the stationary state analysis of the Hopfield model. Namely, the transient dynamics given by the signal-to-noise analysis gives an exact solution in both the stationary state and the first few time steps in the dynamics.

Preliminary Definitions

Let us consider a system storing an infinite number of limit cycles with l finite steps. The system consists of N *Ising-spin neurons* $\sigma_i = \pm 1$. We consider the case where $N \rightarrow \infty$. The spins update their state synchronously with the probability [MKO04]:

$$\begin{aligned} \Pr[\sigma_i(t+1) = -\sigma_i(t)] &= \frac{1}{2}[1 - \sigma_i(t) \tanh \beta h_i(t)], \\ h_i(t) &= J_{ij}\sigma_j(t) + \theta_i(t), \end{aligned} \quad (3.393)$$

where β is the inverse temperature, $\beta = 1/T$. When the temperature is $T = 0$, the updating rule of the state is deterministic. The term $\theta_i(t)$ is a time-dependent external field which is introduced in order to define a response function. The interaction J_{ij} stores p random patterns $\boldsymbol{\xi}^{\nu,\mu} = (\xi_1^{\nu,\mu}, \dots, \xi_N^{\nu,\mu})^T$ so as to retrieve the patterns as:

$$\boldsymbol{\xi}^{\nu,1} \rightarrow \boldsymbol{\xi}^{\nu,2} \rightarrow \dots \rightarrow \boldsymbol{\xi}^{\nu,l} \rightarrow \boldsymbol{\xi}^{\nu,1},$$

sequentially for any μ th *limit cycle*. For instance, the entries of the *synnaptic interaction matrix* $\mathbf{J} = (J_{ij})$ are given by

$$J_{ij} = \frac{1}{N} \xi_i^{\nu,\mu+1} \xi_j^{\nu,\mu},$$

where the pattern index μ are understood to be taken modulo l . Since the number of limit cycles is p/l , the total number of stored patterns is p . The number of stored patterns p is given by $p = \alpha N$, where α is called the loading rate. In our analysis, the number of steps for each limit cycle l is kept finite. Each component of the patterns is assumed to be an independent random variable that takes a value of either $+1$ or -1 according to the probability:

$$\Pr[\xi_i^{\nu,\mu} = \pm 1] = \frac{1}{2}.$$

For the subsequent analysis, the matrix $\mathbf{J}$ is represented as

$$\mathbf{J} = \frac{1}{N} \boldsymbol{\xi}^T \mathbf{S} \boldsymbol{\xi},$$

where the $p \times N$ matrix $\boldsymbol{\xi}$ is defined by

$$\boldsymbol{\xi} = (\xi^{1,1} \dots \xi^{1,l} \xi^{2,1} \dots \xi^{2,l} \dots \xi^{p/l,1} \dots \xi^{p/l,l})^T,$$

and the $p \times p$ matrix $\mathbf{S}$ is defined by

$$\mathbf{S} = \begin{pmatrix} \mathbf{S}' & \dots & 0 \\ & \ddots & \\ 0 & \dots & \mathbf{S}' \end{pmatrix},$$

and the $l \times l$ matrix is defined by $\mathbf{S}' = (S'_{\mu\nu}) = (\delta_{\mu,(\nu+1) \bmod l})$. When $l = 1$, i.e., $\mathbf{S} = \mathbf{1}$ ($\mathbf{1}$ is the unity matrix), the system is equivalent to the Hopfield model.

Path-Integral Analysis

Düring *et al.* discussed the sequential associative memory model by means of the path-integral analysis [DCS98]. In this section, we introduce macroscopic state equations for the model with a finite temperature $T \geq 0$, according to their paper.

In order to analyze the transient dynamics, the generating function $Z[\boldsymbol{\psi}]$ is defined by

$$Z[\boldsymbol{\psi}] = \sum_{\boldsymbol{\sigma}(0), \dots, \boldsymbol{\sigma}(t)} p[\boldsymbol{\sigma}(0), \dots, \boldsymbol{\sigma}(t)] e^{-i \sum_{s < t} \boldsymbol{\sigma}(s) \cdot \boldsymbol{\psi}(s)}, \quad (3.394)$$

where $\boldsymbol{\psi} = (\psi(0), \dots, \psi(t-1))^T$ and the state $\boldsymbol{\sigma}(s) = (\sigma_1(s), \dots, \sigma_N(s))^T$ denotes the state of the spins at time s . The probability $p[\boldsymbol{\sigma}(0), \dots, \boldsymbol{\sigma}(t)]$ denotes the probability of taking the path from initial state $\boldsymbol{\sigma}(0)$ to state $\boldsymbol{\sigma}(t)$ at time t through $\boldsymbol{\sigma}(1), \boldsymbol{\sigma}(2), \dots, \boldsymbol{\sigma}(t-1)$. As (3.394) shows, the generating functional entails the summation of all $2^{(t+1)N}$ paths which the system can take from time 0 to t . One can get all the relevant order parameters, i.e.,

the overlap $m(s)$, the correlation function $C(s, s')$ and the response function $G(s, s')$, by calculating the appropriate derivatives of the above functional and letting $\boldsymbol{\psi}$ tend to $\mathbf{0}$ afterwards as:

$$m(s) = i \lim_{\boldsymbol{\psi} \rightarrow \mathbf{0}} \frac{1}{N} \sum_{i=1}^N \xi_i^{1,s} \frac{\partial Z[\boldsymbol{\psi}]}{\partial \psi_i(s)}, \quad (3.395)$$

$$C(s, s') = - \lim_{\boldsymbol{\psi} \rightarrow \mathbf{0}} \frac{1}{N} \sum_{i=1}^N \frac{\partial^2 Z[\boldsymbol{\psi}]}{\partial \psi_i(s) \partial \psi_i(s')}, \quad (3.396)$$

$$G(s, s') = i \lim_{\boldsymbol{\psi} \rightarrow \mathbf{0}} \frac{1}{N} \sum_{i=1}^N \frac{\partial^2 Z[\boldsymbol{\psi}]}{\partial \psi_i(s) \partial \theta_i(s')}. \quad (3.397)$$

Using the assumption of self-averaging, we replace the generating functional $Z[\boldsymbol{\psi}]$ with its *disorder-averaged generating functional* $\bar{Z}[\boldsymbol{\psi}]$. Evaluating the averaged generating function $\bar{Z}[\boldsymbol{\psi}]$ through the saddle point method, we get the following saddle-point equations for the order parameters of (3.395)–(3.397) in the *thermodynamic limit* (i.e., $N \rightarrow \infty$),

$$m(s) = \ll \xi \sigma(s) \gg, \quad (3.398)$$

$$C(s, s') = \ll \sigma(s) \sigma(s') \gg, \quad (3.399)$$

$$G(s, s') = \frac{\partial \ll \sigma(s) \gg}{\partial \theta(s')}. \quad (3.400)$$

The average over the *effective path measure* is now given by

$$\ll g(\boldsymbol{\sigma}, \mathbf{v}) \gg = \left\langle \int \mathcal{D}\mathbf{v} \operatorname{Tr}_{\boldsymbol{\sigma}} g(\boldsymbol{\sigma}, \mathbf{v}) p[\sigma(0)] \prod_{s=1}^t \frac{1}{2} [1 + \sigma(s) \tanh \beta h(\boldsymbol{\sigma}, \mathbf{v}, s-1)] \right\rangle_{\boldsymbol{\xi}}, \quad (3.401)$$

where $\mathcal{D}\mathbf{v} \equiv \frac{d\mathbf{v} e^{-\frac{1}{2} \mathbf{v} \cdot \mathbf{R}^{-1} \mathbf{v}}}{\sqrt{(2\pi)^t |\mathbf{R}|}}, \quad \operatorname{Tr}_{\boldsymbol{\sigma}} \equiv \sum_{\sigma(0), \dots, \sigma(t) \in \{-1, 1\}}$,

$$h(\boldsymbol{\sigma}, \mathbf{v}, s) = \xi^{s+1} m(s) + \theta(s) + \sqrt{\alpha} v(s) + (\boldsymbol{\Gamma} \boldsymbol{\sigma})(s), \quad (3.402)$$

$$\mathbf{R} = \sum_{a=0}^{l-1} \sum_{m, n \geq 0} \mathbf{G}^{ml+a} \mathbf{C}(\mathbf{G}^\dagger)^{nl+a},$$

$$\boldsymbol{\Gamma} = \frac{\alpha}{l} \sum_{\mu=0}^{l-1} e^{2\pi i \mu / l} [\mathbf{1} - e^{2\pi i \mu / l} \mathbf{G}]^{-1}, \quad \text{with} \quad (3.403)$$

$$\boldsymbol{\Gamma} = \hat{\mathbf{K}}^\dagger, \quad \hat{\mathbf{Q}} = -\frac{1}{2} \alpha i \hat{\mathbf{R}} \quad \text{and} \quad p[\sigma(0)] = \frac{1}{2} [1 + \sigma(0) m(0)],$$

which is the initial spin probability. The operator $\langle \cdot \rangle_{\boldsymbol{\xi}}$ denotes the average over the condensed patterns. The term $(\boldsymbol{\Gamma} \boldsymbol{\sigma})(s)$ denotes the s th element of the vector $\boldsymbol{\Gamma} \boldsymbol{\sigma}$. The vectors $\boldsymbol{\sigma}$ and $\mathbf{v}$ are $\boldsymbol{\sigma} = \{\sigma(0), \dots, \sigma(t)\}$ and

$\mathbf{v} = \{v(0), \dots, v(t-1)\}$, respectively. Equations (3.398)–(3.403) entirely describe the dynamics of the system. The product-term

$$\prod_{s=1}^t \frac{1}{2} [1 + \sigma(s) \tanh \beta h(\boldsymbol{\sigma}, \mathbf{v}, s-1)]$$

in (3.401) cannot be factorized with respect to spin variables at different times. Calculation of the spin summation of (3.401) requires an exponential time $\mathcal{O}(e^t)$ at time t . In the infinite-step sequence processing model, local field $h(\boldsymbol{\sigma}, \mathbf{v}, s)$ depends on only spin variables at time s [DCS98]. Therefore the above product-term can be factorized, so the spin summations can be calculated.

The Stationary State

In this section, we inspect time-translation invariant solutions of our macroscopic equations (3.398)–(3.400) for the deterministic dynamics, i.e., $\beta \rightarrow \infty$ ($T = 0$). The time-translation invariant solutions will describe motion on a macroscopic limit cycle:

$$\begin{cases} m(s) = m, \\ C(s, s') = C(s - s'), \\ G(s, s') = G(s - s'), \end{cases} \quad (3.404)$$

with $\theta(s) = \theta$. Now, we disregard the transient states. Note that the condition of (3.404) includes an unspoken condition that the transient states are disregarded. Therefore, we put that the dynamics is already in the stationary state at time $s = 0$ under this assumption. In the zero noise limit, i.e., $T = 0$ ($\beta \rightarrow \infty$), the dynamics becomes deterministic. Therefore, we also assume that the system takes a fixed path as

$$\sigma(s+l) = \sigma(s),$$

for any time $s \geq 0$. The path which the system takes after time $s \geq 0$ can be described as

$$\sigma(s) = \eta_s, \quad (3.405)$$

by only l constants $\eta_0, \dots, \eta_{l-1} \in \{-1, 1\}$. The pattern index s of the constants η_s is understood to be taken modulo l . Note that it is not necessary to calculate these constants $\{\eta_s\}$ explicitly. When the variable transformation

$$\chi(s) = \eta_s \sigma(s), \quad (3.406)$$

is carried out to spin variables $\sigma(s)$, the transformed spin variables $\chi(s)$ take same value for any time s , i.e., $\chi(s) = \chi(s')$ for any s, s' . Equation (3.401) means the expectation of $g(\boldsymbol{\sigma}, \mathbf{v})$ with respect to the path probability. In the zero noise limit, equation (3.401) becomes

$$\ll g(\boldsymbol{\sigma}, \mathbf{v}) \gg = \left\langle \int \mathcal{D}\mathbf{v} \operatorname{Tr}_{\boldsymbol{\sigma}} g(\boldsymbol{\sigma}, \mathbf{v}) p[\boldsymbol{\sigma}(0)] \prod_{s=1}^t \delta_{\boldsymbol{\sigma}(s), \operatorname{sgn} h(\boldsymbol{\sigma}, \mathbf{v}, s-1)} \right\rangle_{\boldsymbol{\xi}}, \quad (3.407)$$

When the spin variables have the *periodicity*: $\sigma(s+l) = \sigma(s)$, the *Gaussian random fields* are also deterministic: $v(s+l) = v(s)$. For any function $\phi(\{\sigma(s)\})$ and any constants $c_0, \dots, c_t \in \{-1, 1\}$, the following identity holds:

$$\sum_{\sigma(0), \dots, \sigma(t) \in \{-1, 1\}} \phi(\sigma(0), \dots, \sigma(t)) = \sum_{\sigma(0), \dots, \sigma(t) \in \{-1, 1\}} \phi(c_0 \sigma(0), \dots, c_t \sigma(t)). \quad (3.408)$$

Applying (3.408) to (3.407) and substituting (3.406), we get [MKO04]

$$\begin{aligned} \ll g(\boldsymbol{\sigma}, \mathbf{v}) \gg &= \left\langle \int \mathcal{D}\mathbf{v} \operatorname{Tr}_{\boldsymbol{\sigma}} g(\{c_0 \sigma(0), \dots, c_t \sigma(t)\}, \mathbf{v}) p[c_0 \sigma(0)] \right. \\ &\quad \times \left. \prod_{s=1}^t \delta_{c_s \sigma(s), \operatorname{sgn} h(\{c_0 \sigma(0), \dots, c_t \sigma(t)\}, \mathbf{v}, s-1)} \right\rangle_{\boldsymbol{\xi}} \\ &= \left\langle \int \mathcal{D}\mathbf{v} \operatorname{Tr}_{\boldsymbol{\chi}} g(\{c_0 \eta_0 \chi(0), \dots, c_t \eta_t \chi(t)\}, \mathbf{v}) \right. \\ &\quad \times \left. p[c_0 \eta_0 \chi(0)] \prod_{s=1}^t \delta_{c_s \eta_s \chi(s), \operatorname{sgn} h(\{c_0 \eta_0 \chi(0), \dots, c_t \eta_t \chi(t)\}, \mathbf{v}, s-1)} \right\rangle_{\boldsymbol{\xi}} = \\ &= \left\langle \int \mathcal{D}\mathbf{v} \operatorname{Tr}_{\boldsymbol{\chi}} g(\{\chi(0), \dots, \chi(t)\}, \mathbf{v}) p[\chi(0)] \prod_{s=1}^t \delta_{\chi(s), \operatorname{sgn} h(\{\chi(0), \dots, \chi(t)\}, \mathbf{v}, s-1)} \right\rangle_{\boldsymbol{\xi}}, \\ &\text{with } \operatorname{Tr}_{\boldsymbol{\chi}} \equiv \sum_{\eta_0 \chi(0), \dots, \eta_t \chi(t) \in \{-1, 1\}} = \sum_{\chi(0), \dots, \chi(t) \in \{-1, 1\}}. \end{aligned} \quad (3.409)$$

In (3.409) we have put the constants $\{c_s\}$ as $c_s = \eta_s$ for all s . Generality is kept even if $c_s = \eta_s$ for all s . Namely, with respect to the transformed spin variable $\chi(s)$, the effective single spin described by (3.409) is

$$\chi(s) = \operatorname{sgn} h(\{\chi(0), \dots, \chi(t)\}, \mathbf{v}, s-1). \quad (3.410)$$

The transformed spin variables $\chi(s)$ are deterministic even if (3.410) includes the Gaussian random fields $\mathbf{v}$, since the Gaussian random fields are deterministic. In order to get rid of the self-interaction, we assume the Maxwell construction ansatz. Using the identity $\chi(s) = \chi(s')$ for any s, s' to (3.410) and applying the Maxwell construction, we get

$$\begin{aligned} \chi(s) &= \operatorname{sgn} [\xi^s m(s-1) + \theta(s-1) + \sqrt{\alpha} v(s-1) + \sum_{s' < s} \Gamma(s, s') \chi(s')] \\ &= \operatorname{sgn} [\xi^s m(s-1) + \theta(s-1) + \sqrt{\alpha} v(s-1) + \chi(s) \sum_{s' < s} \Gamma(s, s')] \\ &= \operatorname{sgn} h(\mathbf{v}, s-1), \\ &\text{with } h(\mathbf{v}, s) \equiv \xi^{s+1} m(s) + \theta(s) + \sqrt{\alpha} v(s). \end{aligned} \quad (3.411)$$

Substituting (3.411) into (3.409), we get

$$\ll g(\boldsymbol{\sigma}, \mathbf{v}) \gg = \left\langle \int \mathcal{D}\mathbf{v} \operatorname{Tr}_{\boldsymbol{\chi}} g(\boldsymbol{\chi}, \mathbf{v}) p[\chi(0)] \prod_{s=1}^t \delta_{\boldsymbol{\chi}, \operatorname{sgn} h(\mathbf{v}, s-1)} \right\rangle_{\boldsymbol{\xi}}. \quad (3.412)$$

Thus, we can get rid of the self-interaction in the single spin problem by using the Maxwell construction in the zero noise limit, i.e., $T = 0$. Since (3.412) can be factorized with respect to the transformed spin variables $\chi(s)$ at different times, we can easily perform the spin summations. After simple rescalings we arrive at

$$m(s) = \left\langle \xi^s \int \mathcal{D}\mathbf{v} \operatorname{sgn} h(\mathbf{v}, s-1) \right\rangle_{\boldsymbol{\xi}}, \quad (3.413)$$

$$C(s, s') = \delta_{s, s'} + [1 - \delta_{s, s'}] \left\langle \int \mathcal{D}\mathbf{v} [\operatorname{sgn} h(\mathbf{v}, s-1)] [\operatorname{sgn} h(\mathbf{v}, s'-1)] \right\rangle_{\boldsymbol{\xi}},$$

$$G(s, s') = \delta_{s, s'-1} \lim_{\beta \rightarrow \infty} \beta \left\{ 1 - \left\langle \int \mathcal{D}\mathbf{v} \tanh^2 \beta h(\mathbf{v}, s-1) \right\rangle_{\boldsymbol{\xi}} \right\}. \quad (3.414)$$

We now calculate the matrix $\mathbf{R}$ under the condition of (3.404). Since the matrices $\mathbf{G}$ and $\mathbf{C}$ become *Toeplitz matrices* (in particular, $\mathbf{C}$ is symmetric) under this conditions, $\mathbf{C}$ and $\mathbf{G}$ can be approximately regarded as commuting matrices, i.e., $\mathbf{CG} = \mathbf{GC}$. Therefore, the matrix $\mathbf{R}$ simplifies to

$$\begin{aligned} \mathbf{R} &= \sum_{a=0}^{l-1} \mathbf{G}^a \left(\sum_{m, n \geq 0} \mathbf{G}^{ml} (\mathbf{G}^\dagger)^{nl} \mathbf{C} \right) (\mathbf{G}^\dagger)^a \\ &= \sum_{a=0}^{l-1} \mathbf{G}^a ([\mathbf{1} - \mathbf{G}^l]^{-1} [\mathbf{1} - (\mathbf{G}^\dagger)^l]^{-1} \mathbf{C}) (\mathbf{G}^\dagger)^a \\ &= [\mathbf{1} - \mathbf{GG}^\dagger]^{-1} [\mathbf{1} - (\mathbf{GG}^\dagger)^l] [\mathbf{1} - \mathbf{G}^l]^{-1} [\mathbf{1} - (\mathbf{G}^\dagger)^l]^{-1} \mathbf{C}, \end{aligned} \quad (3.415)$$

We consider the persistent parts of $C(\tau)$ and $R(\tau)$ as $C(\tau) \rightarrow q$ and $R(\tau) \rightarrow r$ for $\tau \rightarrow \infty$ and also consider the non-persistent parts $\tilde{C}(\tau) \rightarrow 0$ and $\tilde{R}(\tau) \rightarrow 0$, i.e.,

$$C(\tau) = q + \tilde{C}(\tau) \quad \text{and} \quad R(\tau) = r + \tilde{R}(\tau).$$

Upon rewriting $G(\tau) = \beta \delta_{\tau, 1} [1 - \tilde{q}]$ and $r = q\rho$ using (3.415), from (3.413)–(3.414) we get

$$\begin{aligned} m &= \left\langle \xi \int Dz \int \tilde{\mathcal{D}}\mathbf{v} \operatorname{sgn} h(\mathbf{v}, 0|z) \right\rangle_{\boldsymbol{\xi}}, \\ q &= \left\langle \int Dz \int \tilde{\mathcal{D}}\mathbf{v} [\operatorname{sgn} h(\mathbf{v}, \tau|z)] [\operatorname{sgn} h(\mathbf{v}, 0|z)] \right\rangle_{\boldsymbol{\xi}}, \\ \tilde{q} &= \lim_{\beta \rightarrow \infty} \left\langle \int Dz \int \tilde{\mathcal{D}}\mathbf{v} \tanh^2 \beta h(\mathbf{v}, 0|z) \right\rangle_{\boldsymbol{\xi}}, \\ r &= q\rho, \quad \text{with} \end{aligned}$$

$$\tilde{D}\mathbf{v} \equiv d\mathbf{v}e^{-\frac{1}{2}\mathbf{v}\cdot\tilde{\mathbf{R}}^{-1}\mathbf{v}}[(2\pi)^t|\tilde{\mathbf{R}}|]^{-1/2}, \quad Dz = \frac{dz}{\sqrt{2\pi}}e^{-z^2/2} \quad \text{and}$$

$$h(\mathbf{v}, \tau|z) \equiv \xi^\tau m + \theta + z\sqrt{\alpha q\rho} + \sqrt{\alpha}v(\tau).$$

The matrix $\mathbf{G}$ is, from (3.414), given by

$$\mathbf{G} = \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 \\ G(1) & 0 & 0 & \cdots & 0 \\ 0 & G(1) & 0 & \cdots & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & \cdots & 0 & G(1) & 0 \end{pmatrix}.$$

Equation (3.415) can be changed to

$$[\mathbf{1} - (\mathbf{G}^\dagger)^l - \mathbf{G}^l - (\mathbf{G}^\dagger \mathbf{G})^l][\mathbf{1} - (\mathbf{G} \mathbf{G}^\dagger)^l]^{-1}[\mathbf{1} - \mathbf{G} \mathbf{G}^\dagger] \mathbf{R} = \mathbf{C}. \quad (3.416)$$

The identity $\mathbf{G} \mathbf{G}^\dagger \simeq G(1)^2 \mathbf{1}$ holds in the case where time s is sufficiently large, so the left-hand side of (3.416) becomes

$$\begin{aligned} & [\mathbf{1} - (\mathbf{G}^\dagger)^l - \mathbf{G}^l - (\mathbf{G}^\dagger \mathbf{G})^l][\mathbf{1} - (\mathbf{G} \mathbf{G}^\dagger)^l]^{-1}[\mathbf{1} - \mathbf{G} \mathbf{G}^\dagger] \mathbf{R} \\ &= \begin{pmatrix} D(0) & D(1) & \cdots & D(t-1) \\ D(1) & D(0) & \cdots & D(t-2) \\ \vdots & \vdots & \ddots & \vdots \\ D(t-1) & D(t-2) & \cdots & D(0) \end{pmatrix}, \quad \text{where} \end{aligned}$$

$$\begin{aligned} D(0) &= (1 + g_1^2)g_2 R(0) - 2g_1 g_2 R(1), \\ D(s) &= (1 + g_1^2)g_2 R(s) - g_1 g_2 [R(s-1) + R(s+1)], \quad (0 < s < t-1), \\ D(t-1) &= (1 + g_1^2)g_2 R(t-1) - 2g_1 g_2 R(t-2), \quad \text{with} \\ g_1 &\equiv G(1)^l \quad \text{and} \quad g_2 \equiv [1 - G(1)^2]/[1 - G(1)^{2l}]. \end{aligned}$$

Since $\mathbf{D}$ and $\mathbf{C}$ are symmetric Toeplitz matrices, they can be diagonalized by using the discrete Fourier transformation [MKO04]. The *Fourier transformations* (or, the *Lattice Green's functions*) $\hat{D}_k, \hat{C}_k$ of the matrices $\mathbf{D}, \mathbf{C}$ are given by

$$\begin{aligned} \hat{D}_k &\simeq \sum_{\tau=0}^{t-1} \{(1 + g_1^2)g_2 R(\tau) - g_1 g_2 [R(\tau-1) - R(\tau+1)]\} e^{ik\tau}, \\ \hat{C}_k &= \sum_{\tau=0}^{t-1} C(\tau) e^{ik\tau}. \end{aligned}$$

For any wave number k , $\hat{D}_k = \hat{C}_k$ holds when $\mathbf{D} = \mathbf{C}$. Taking the limit $s \rightarrow \infty$ about $\hat{D}_0 = \hat{C}_0$, the following relationship is obtained:

$$r = \frac{1 - G(1)^{2l}}{(1 - G(1)^2)(1 - G(1)^l)^2} q.$$

By working out the remaining integrals over $\mathbf{v}$ and setting $\theta = 0$, we finally get the stationary state equations of the order parameters as follows:

$$m = \operatorname{erf} \left(\frac{m}{\sqrt{2\alpha\rho}} \right), \quad (3.417)$$

$$U = \sqrt{\frac{2}{\pi\alpha\rho}} e^{-\frac{m^2}{2\alpha\rho}},$$

$$\rho = \frac{1 - U^{2l}}{(1 - U^2)(1 - U^l)^2}, \quad (3.418)$$

with $q = 1$, $\tilde{q} = 1$ and $U \equiv G(1)$ where $\operatorname{erf}(\cdot)$ denotes the *Error function*

$$\operatorname{erf}(x) \equiv \frac{2}{\pi} \int_0^x e^{-u^2} du.$$

It turns out that these stationary state equations (3.417)–(3.418) given by this exact solution are equivalent to those of the signal-to-noise analysis [MO95]. It has been confirmed in [MKO04] that the above theoretical results are in good agreement with the computer simulations. Storage capacity monotonously increases from $\alpha_c = 0.138$ ($l = 1$) with the number of steps l . In the large l limit, storage capacity finally converges to $\alpha_c = 0.269$, which coincides with the theoretical result of for the infinite-step sequence processing model given by Düring *et al.* [DCS98]. The original properties of the finite-step sequence processing model appear as long as the number of steps of a limit cycle l has order $l = \mathcal{O}(1)$. In the case that l has the order more than $\mathcal{O}(1)$, the properties are the same as the properties of the infinite-step sequence processing model. For more details see [MKO04].

3.5.2 General Path-Integral Approach to Attractor NNs

Let us return to the simplest setting in which to study the problem: single pattern recall in an attractor neural network with N binary neurons and $p = \alpha N$ stored patterns in the non-trivial regime, where $\alpha > 0$. We choose parallel dynamics, i.e., (3.295), with Hebbian synapses of the form (3.292) with $A_{\mu\nu} = \delta_{\mu\nu}$, i.e., $J_{ij} = N^{-1} \xi_i^\mu \xi_j^\mu$, giving us the parallel dynamics version of the Hopfield model [Hop82]. Our interest is in the recall overlap $m(\boldsymbol{\sigma}) = N^{-1} \sigma_i \xi_i^1$ between system state and pattern one. We saw above that for $N \rightarrow \infty$ the fluctuations in the values of the recall overlap m will vanish, and that for initial states where all $\sigma_i(0)$ are drawn independently the overlap m will obey:

$$m(t+1) = \int dz P_t(z) \tanh[\beta(m(t) + z)] : \quad (3.419)$$

$$P_t(z) = \lim_{N \rightarrow \infty} \frac{1}{N} \langle \delta[z - \frac{1}{N} \xi_i^1 \xi_i^\mu \xi_j^\mu \sigma_j(t)] \rangle,$$

and that all complications in a dynamical analysis of the $\alpha > 0$ regime are concentrated in the calculation of the distribution $P_t(z)$ of the (generally non-trivial) interference noise.

As a simple *Gaussian approximation* one could just assume [Ama77, AN00] that the σ_i remain uncorrelated at all times, i.e., $\text{Prob}[\sigma_i(t) = \pm \xi_i^1] = \frac{1}{2}[1 \pm m(t)]$ for all $t \geq 0$, such that the argument given above for $t = 0$ (leading to a Gaussian $P(z)$) would hold generally, and where the map (3.419) would describe the overlap evolution at all times:

$$P_t(z) = [2\pi\alpha]^{-\frac{1}{2}} e^{-\frac{1}{2}z^2/\alpha} ;$$

$$m(t+1) = \int Dz \tanh[\beta(m(t) + z\sqrt{\alpha})],$$

with the *Gaussian measure* $Dz = (2\pi)^{-\frac{1}{2}} e^{-\frac{1}{2}z^2} dz$. This equation, however, must be generally incorrect. Rather than taking all σ_i to be independent, a weaker assumption would be to just assume the interference noise distribution $P_t(z)$ to be a zero-average Gaussian one, at any time, with statistically independent noise variables z at different times. One can then derive (for $N \rightarrow \infty$ and fully connected networks) an evolution equation for the width $\Sigma(t)$, giving [AM88]:

$$m(t+1) = \int Dz \tanh[\beta(m(t) + z\Sigma(t))] ;$$

$$P_t(z) = [2\pi\Sigma^2(t)]^{-\frac{1}{2}} e^{-\frac{1}{2}z^2/\Sigma^2(t)} ;$$

$$\Sigma^2(t+1) = \alpha + 2\alpha m(t+1)m(t)h[m(t), \Sigma(t)]$$

$$+ \Sigma^2(t)h^2[m(t), \Sigma(t)], \quad \text{with}$$

$$h[m, \Sigma] = \beta \left[1 - \int Dz \tanh^2[\beta(m + z\Sigma)] \right] .$$

These equations describe correctly the qualitative features of recall dynamics, and are found to work well when retrieval actually occurs. A final refinement of the Gaussian approach [Oka95] consisted in allowing for correlations between the noise variables z at different times (while still describing them by Gaussian distributions). This results in a hierarchy of macroscopic equations, which improve upon the previous Gaussian theories and even predict the correct stationary state and phase diagrams, but still fail to be correct at intermediate times.

In view of the non-Gaussian shape of the interference noise distribution, several attempts have been made at constructing non-Gaussian approximations. In all cases the aim is to arrive at a theory involving only macroscopic observables with a *single* time-argument. For a fully connected network with binary neurons and parallel dynamics a more accurate ansatz for $P_t(z)$ would be the sum of two Gaussian functions. In [HO90] the following choice was proposed:

$$\begin{aligned}
P_t(z) &= P_t^+(z) + P_t^-(z), \\
P_t^\pm(z) &= \lim_{N \rightarrow \infty} \frac{1}{N} \delta_{\sigma_i(t), \pm \xi_i^1} \langle \delta[z - \frac{1}{N} \xi_i^1 \xi_i^\mu \xi_j^\mu \sigma_j(t)] \rangle \\
P_t^\pm(z) &= \frac{1 \pm m(t)}{2\Sigma(t)\sqrt{2\pi}} e^{-\frac{1}{2}[z \mp d(t)]^2 / \Sigma^2(t)},
\end{aligned}$$

followed by a self-consistent calculation of $d(t)$ (representing an effective ‘retarded self-interaction’, since it has an effect equivalent to adding $h_i(\boldsymbol{\sigma}(t)) \rightarrow h_i(\boldsymbol{\sigma}(t)) + d(t)\sigma_i(t)$), and of the width $\Sigma(t)$ of the two distributions $P_t^\pm(z)$, together with

$$\begin{aligned}
m(t+1) &= \frac{1}{2}[1+m(t)] \int Dz \tanh[\beta(m(t)+d(t)+z\Sigma(t))] \\
&+ \frac{1}{2}[1-m(t)] \int Dz \tanh[\beta(m(t)-d(t)+z\Sigma(t))].
\end{aligned}$$

The resulting three-parameter theory, in the form of closed dynamic equations for $\{m, d, \Sigma\}$, is found to give a nice (but not perfect) agreement with numerical simulations.

A different philosophy was followed in [CS94] (for sequential dynamics). First (as yet exact) equations are derived for the evolution of the two macroscopic observables $m(\boldsymbol{\sigma}) = m_1(\boldsymbol{\sigma})$ and $r(\boldsymbol{\sigma}) = \alpha^{-1} \sum_{\mu>1} m_\mu^2(\boldsymbol{\sigma})$, with $m_\mu(\boldsymbol{\sigma}) = N^{-1} \xi_i^1 \sigma_i$, which are both found to involve $P_t(z)$:

$$\begin{aligned}
\dot{m} &= \int dz P_t(z) \tanh[\beta(m+z)], \\
\dot{r} &= \frac{1}{\alpha} \int dz P_t(z) z \tanh[\beta(m+z)] + 1 - r.
\end{aligned}$$

Next one closes these equations *by hand*, using a maximum-entropy (or, ‘Occam’s Razor’) argument: instead of calculating $P_t(z)$ from (3.419) with the real (unknown) microscopic distribution $p_t(\boldsymbol{\sigma})$, it is calculated upon assigning equal probabilities to all states $\boldsymbol{\sigma}$ with $m(\boldsymbol{\sigma}) = m$ and $r(\boldsymbol{\sigma}) = r$, followed by averaging over all realizations of the stored patterns with $\mu > 1$. In order words: one assumes (i) that the microscopic states visited by the system are ‘typical’ within the appropriate (m, r) sub-shells of state space, and (ii) that one can average over the disorder. Assumption (ii) is harmless, the most important step is (i). This procedure results in an explicit (non-Gaussian) expression for the noise distribution in terms of (m, r) only, a closed 2-parameter theory which is exact for short times and in equilibrium, accurate predictions of the macroscopic flow in the (m, r) -plane, but (again) deviations in predicted time-dependencies at intermediate times. This theory, and its performance, was later improved by applying the same ideas to a derivation of a dynamic equation for the function $P_t(z)$ itself (rather than for m and r only).

If we now use the powerful *path-integral formalism* (see [II06b]), instead of working with the probability $p_t(\boldsymbol{\sigma})$ of finding a microscopic state $\boldsymbol{\sigma}$ at time

t in order to calculate the statistics of a set of macroscopic observables $\Omega(\sigma)$ at time t , we turn to the probability $\text{Prob}[\sigma(0), \dots, \sigma(t_m)]$ of finding a microscopic *path* $\sigma(0) \rightarrow \sigma(1) \rightarrow \dots \rightarrow \sigma(t_m)$. We also add time-dependent external sources to the local fields, $h_i(\sigma) \rightarrow h_i(\sigma) + \theta_i(t)$, in order to probe the networks via perturbations and define a response function. The idea is to concentrate on the moment partition function $Z[\psi]$, which, like $\text{Prob}[\sigma(0), \dots, \sigma(t_m)]$, fully captures the statistics of paths:

$$Z[\psi] = \langle e^{-i \sum_{t=0}^{t_m} \psi_i(t) \sigma_i(t)} \rangle.$$

It generates averages of the relevant observables, including those involving neuron states at different times, such as correlation functions $C_{ij}(t, t') = \langle \sigma_i(t) \sigma_j(t') \rangle$ and response functions $G_{ij}(t, t') = \partial \langle \sigma_i(t) \rangle / \partial \theta_j(t')$, upon differentiation with respect to the dummy variables $\{\psi_i(t)\}$:

$$\begin{aligned} \langle \sigma_i(t) \rangle &= i \lim_{\psi \rightarrow 0} \frac{\partial Z[\psi]}{\partial \psi_i(t)}, & C_{ij}(t, t') &= - \lim_{\psi \rightarrow 0} \frac{\partial^2 Z[\psi]}{\partial \psi_i(t) \partial \psi_j(t')}, \\ G_{ij}(t, t') &= i \lim_{\psi \rightarrow 0} \frac{\partial^2 Z[\psi]}{\partial \psi_i(t) \partial \theta_j(t')}. \end{aligned}$$

Next one assumes (correctly) that for $N \rightarrow \infty$ only the statistical properties of the stored patterns will influence the macroscopic quantities, so that the partition function $Z[\psi]$ can be averaged over all pattern realizations, i.e., $Z[\psi] \rightarrow \overline{Z[\psi]}$. As in replica theories (the canonical tool to deal with complexity in equilibrium) one carries out the disorder average *before* the average over the statistics of the neuron states, resulting for $N \rightarrow \infty$ in what can be interpreted as a theory describing a single ‘effective’ binary neuron $\sigma(t)$, with an effective local field $h(t)$ and the dynamics $\text{Prob}[\sigma(t+1) = \pm 1] = \frac{1}{2}[1 \pm \tanh[\beta h(t)]]$. However, this effective local field is found to generally depend on past states of the neuron, and on zero-average but temporally correlated Gaussian noise contributions $\xi(t)$:

$$h(t|\{\sigma\}, \{\xi\}) = m(t) + \theta(t) + \alpha \sum_{t' < t} R(t, t') \sigma(t') + \sqrt{\alpha} \xi(t). \quad (3.420)$$

The first comprehensive neural network studies along these lines, dealing with fully connected networks, were applied to asymmetrically and symmetrically extremely diluted networks [KZ91, WS91]. More recent applications include sequence processing networks [DCS98]¹⁴. For $N \rightarrow \infty$ the differences between different models are found to show up only in the actual form taken by the effective local field (3.420), i.e., in the dependence of the ‘retarded self-interaction’ kernel $R(t, t')$ and the covariance matrix $\langle \xi(t) \xi(t') \rangle$ of the interference-induced Gaussian noise on the macroscopic

¹⁴ In the case of sequence recall the overlap m is defined with respect to the ‘moving’ target, i.e., $m(t) = \frac{1}{N} \sigma_i(t) \xi_i^t$

objects $\mathbf{C} = \{C(s, s') = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_i C_{ii}(s, s')\}$ and $\mathbf{G} = \{G(s, s') = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_i G_{ii}(s, s')\}$. For instance [Coo01, SC00, SC01, CKS05]:

model	synapses J_{ij}	$R(t, t')$	$\langle \xi(t) \xi(t') \rangle$
fully connected, static patterns	$\frac{1}{N} \xi_i^\mu \xi_j^\mu$	$[(\mathbf{1} - \mathbf{G})^{-1} \mathbf{G}](t, t')$	$[(\mathbf{1} - \mathbf{G})^{-1} \mathbf{C} (\mathbf{1} - \mathbf{G}^\dagger)^{-1}](t, t')$
fully connected, pattern sequence	$\frac{1}{N} \xi_i^{\mu+1} \xi_j^\mu$	0	$\sum_{n \geq 0} [(\mathbf{G}^\dagger)^n \mathbf{C} \mathbf{G}^n](t, t')$
symm extr diluted, static patterns	$\frac{c_{ij}}{c} \xi_i^\mu \xi_j^\mu$	$G(t, t')$	$C(t, t')$
asymm extr diluted, static patterns	$\frac{c_{ij}}{c} \xi_i^\mu \xi_j^\mu$	0	$C(t, t')$

with the c_{ij} drawn at random according to $P(c_{ij}) = \frac{c}{N} \delta_{c_{ij}, 1} + (1 - \frac{c}{N}) \delta_{c_{ij}, 0}$ (either symmetrically, i.e., $c_{ij} = c_{ji}$, or independently) and where $c_{ii} = 0$, $\lim_{N \rightarrow \infty} c/N = 0$, and $c \rightarrow \infty$. In all cases the observables (overlaps and correlation- and response-functions) are to be solved from the following closed equations, involving the statistics of the single effective neuron experiencing the field (3.420):

$$m(t) = \langle \sigma(t) \rangle, \quad C(t, t') = \langle \sigma(t) \sigma(t') \rangle, \quad G(t, t') = \partial \langle \sigma(t) \rangle / \partial \theta(t').$$

It is now clear that Gaussian theories can at most produce exact results for asymmetric networks. Any degree of symmetry in the synapses is found to induce a non-zero retarded self-interaction, via the kernel $K(t, t')$, which constitutes a non-Gaussian contribution to the local fields. Exact closed macroscopic theories apparently require a number of macroscopic observables which grows as $\mathcal{O}(t^2)$ in order to predict the dynamics up to time t . In the case of sequential dynamics the picture is found to be very similar to the one above; instead of discrete time labels $t \in \{0, 1, \dots, t_m\}$, path summations and matrices, there one has a real time variable $t \in [0, t_m]$, path-integrals, integral operators, and partition-functions.

Partition-Function Analysis for Binary Neurons

First we will define parallel dynamics, i.e., (3.295), driven as usual by local fields of the form $h_i(\boldsymbol{\sigma}; t) = J_{ij} \sigma_j + \theta_i(t)$, but with a more general choice of *Hebbian synapses*, in which we allow for a possible random dilution (to reduce repetition in our subsequent derivations):

$$J_{ij} = \frac{c_{ij}}{c} \xi_i^\mu \xi_j^\mu, \quad \text{with} \quad p = \alpha c. \quad (3.421)$$

Architectural properties are reflected in the variables $c_{ij} \in \{0, 1\}$, whereas information storage is to be effected by the remainder in (3.421), involving p randomly and independently drawn patterns $\boldsymbol{\xi}^\mu = (\xi_1^\mu, \dots, \xi_N^\mu) \in \{-1, 1\}^N$.

I will deal both with symmetric and with asymmetric architectures (always putting $c_{ii} = 0$), in which the variables c_{ij} are drawn randomly according to

$$\begin{aligned} c_{ij} &= c_{ji}, & (\text{for all } i < j), & \quad P(c_{ij}) = \frac{c}{N} \delta_{c_{ij},1} + (1 - \frac{c}{N}) \delta_{c_{ij},0}, \\ (\text{for all } i &= j), & \quad P(c_{ij}) = \frac{c}{N} \delta_{c_{ij},1} + (1 - \frac{c}{N}) \delta_{c_{ij},0}. \end{aligned}$$

Thus c_{kl} is statistically independent of c_{ij} as soon as $(k, l) \notin \{(i, j), (j, i)\}$. In leading order in N one has $\langle \sum_j c_{ij} \rangle = c$ for all i , so c gives the average number of neurons contributing to the field of any given neuron. In view of this, the number p of patterns to be stored can be expected to scale as $p = \alpha c$. The connectivity parameter c is chosen to diverge with N , i.e., $\lim_{N \rightarrow \infty} c^{-1} = 0$. If $c = N$ we get the fully connected (parallel dynamics) Hopfield model. Extremely diluted networks are got when $\lim_{N \rightarrow \infty} c/N = 0$.

For simplicity, we make the so-called ‘condensed ansatz’: we assume that the system state has an $\mathcal{O}(N^0)$ overlap only with a single pattern, say $\mu = 1$. This situation is induced by initial conditions: we take a randomly drawn $\sigma(0)$, generated by

$$p(\sigma(0)) = \prod_i \left\{ \frac{1}{2} [1 + m_0] \delta_{\sigma_i(0), \xi_i^1} + \frac{1}{2} [1 - m_0] \delta_{\sigma_i(0), -\xi_i^1} \right\},$$

$$\text{so} \quad \frac{1}{N} \xi_i^1 \langle \sigma_i(0) \rangle = m_0.$$

The patterns $\mu > 1$, as well as the architecture variables c_{ij} , are viewed as disorder. One assumes that for $N \rightarrow \infty$ the macroscopic behavior of the system is ‘self-averaging’, i.e., only dependent on the statistical properties of the disorder (rather than on its microscopic realization). Averages over the disorder are written as $\overline{}$. We next define the *disorder-averaged partition function*:

$$\overline{Z[\psi]} = \overline{\langle e^{-i \sum_t \psi_i(t) \sigma_i(t)} \rangle}, \quad (3.422)$$

in which the time t runs from $t = 0$ to some (finite) upper limit t_m . Note that $\overline{Z[0]} = 1$. With a modest amount of foresight we define the macroscopic site-averaged and disorder-averaged objects $m(t) = N^{-1} \xi_i^1 \langle \sigma_i(t) \rangle$, $C(t, t') = N^{-1} \langle \sigma_i(t) \sigma_i(t') \rangle$ and $G(t, t') = N^{-1} \partial \langle \sigma_i(t) \rangle / \partial \theta_i(t')$. They can be obtained from (3.422) as follows:

$$\begin{aligned} m(t) &= \lim_{\psi \rightarrow 0} \frac{i}{N} \xi_j^1 \frac{\partial \overline{Z[\psi]}}{\partial \psi_j(t)}, & C(t, t') &= - \lim_{\psi \rightarrow 0} \frac{1}{N} \frac{\partial^2 \overline{Z[\psi]}}{\partial \psi_j(t) \partial \psi_j(t')}, \\ G(t, t') &= \lim_{\psi \rightarrow 0} \frac{i}{N} \frac{\partial^2 \overline{Z[\psi]}}{\partial \psi_j(t) \partial \theta_j(t')}. \end{aligned}$$

Now, as in equilibrium replica calculations, the hope here is that progress can be made by carrying out the disorder averages first. In equilibrium calculations we use the replica trick to convert our disorder averages into feasible

ones; here the idea is to isolate the local fields at different times and different sites by inserting appropriate δ -distributions:

$$\begin{aligned} 1 &= \prod_{it} \int dh_i(t) \delta[h_i(t) - J_{ij}\sigma_j(t) - \theta_i(t)] \\ &= \int \{d\mathbf{h}d\hat{\mathbf{h}}\} e^{i \sum_{it} \hat{h}_i(t)[h_i(t) - J_{ij}\sigma_j(t) - \theta_i(t)]}, \end{aligned}$$

with $\{d\mathbf{h}d\hat{\mathbf{h}}\} = \prod_{it} [d\hat{h}_i(t)dh_i(t)/2\pi]$, giving

$$\begin{aligned} \overline{Z[\psi]} &= \int \{d\mathbf{h}d\hat{\mathbf{h}}\} e^{i \sum_{it} \hat{h}_i(t)[h_i(t) - \theta_i(t)]} \\ &\quad \times \langle e^{-i \sum_{it} \psi_i(t)\sigma_i(t)} \left[e^{-i \sum_{it} \hat{h}_i(t)J_{ij}\sigma_j(t)} \right] \rangle_{\text{pf}}, \end{aligned}$$

in which $\langle \dots \rangle_{\text{pf}}$ refers to averages over a constrained stochastic process of the type (3.295), but with prescribed fields $\{h_i(t)\}$ at all sites and at all times. Note that with such prescribed fields the probability of partition a path $\{\sigma(0), \dots, \sigma(t_m)\}$ is given by

$$\begin{aligned} \text{Prob}[\sigma(0), \dots, \sigma(t_m) | \{h_i(t)\}] &= \\ p(\sigma(0)) e^{\sum_{it} [\beta\sigma_i(t+1)h_i(t) - \log 2 \cosh[\beta h_i(t)]]}, &\quad \text{so} \\ \overline{Z[\psi]} &= \int \{d\mathbf{h}d\hat{\mathbf{h}}\} \sum_{\sigma(0)} \dots \sum_{\sigma(t_m)} p(\sigma(0)) e^{N\mathcal{F}[\{\sigma\}, \{\hat{\mathbf{h}}\}]} \\ &\quad \times \prod_{it} e^{i\hat{h}_i(t)[h_i(t) - \theta_i(t)] - i\psi_i(t)\sigma_i(t) + \beta\sigma_i(t+1)h_i(t) - \log 2 \cosh[\beta h_i(t)]}, \\ &\quad \text{with} \quad \mathcal{F}[\{\sigma\}, \{\hat{\mathbf{h}}\}] = \frac{1}{N} \log \left[e^{-i \sum_{it} \hat{h}_i(t)J_{ij}\sigma_j(t)} \right]. \end{aligned} \quad (3.423)$$

We concentrate on the term $\mathcal{F}[\dots]$ (with the disorder), of which we need only know the limit $N \rightarrow \infty$, since only terms inside $\overline{Z[\psi]}$ which are exponential in N will retain statistical relevance. In the disorder-average of (3.423) every site i plays an equivalent role, so the leading order in N of (3.423) should depend only on site-averaged functions of the $\{\sigma_i(t), \hat{h}_i(t)\}$, with no reference to any special direction except the one defined by pattern ξ^1 . The simplest such functions with a single time variable are

$$a(t; \{\sigma\}) = \frac{1}{N} \xi_i^1 \sigma_i(t), \quad k(t; \{\hat{\mathbf{h}}\}) = \frac{1}{N} \xi_i^1 \hat{h}_i(t),$$

whereas the simplest ones with two time variables would appear to be

$$\begin{aligned} q(t, t'; \{\sigma\}) &= \frac{1}{N} \sigma_i(t) \sigma_i(t'), & Q(t, t'; \{\hat{\mathbf{h}}\}) &= \frac{1}{N} \hat{h}_i(t) \hat{h}_i(t'), \\ K(t, t'; \{\sigma, \hat{\mathbf{h}}\}) &= \frac{1}{N} \hat{h}_i(t) \sigma_i(t'). \end{aligned}$$

It will turn out that all models of the type (3.421), have the crucial property that above are in fact the *only* functions to appear in the leading order of (3.423):

$$\mathcal{F}[\dots] = \Phi[\{a(t; \dots), k(t; \dots), q(t, t'; \dots), Q(t, t'; \dots), K(t, t'; \dots)\}] + \dots,$$

for $N \rightarrow \infty$ and some as yet unknown function $\Phi[\dots]$. This allows us to proceed with the evaluation of $\bar{Z}[\psi]$. We can introduce suitable δ -distributions (taking care that all exponents scale linearly with N , to secure statistical relevance). Thus we insert

$$\begin{aligned} 1 &= \prod_{t=0}^{t_m} \int da(t) \delta[a(t) - a(t; \{\sigma\})] \\ &= \left[\frac{N}{2\pi} \right]^{t_m+1} \int d\mathbf{a} d\hat{\mathbf{a}} e^{iN \sum_t \hat{a}(t)[a(t) - \frac{1}{N} \xi_j^1 \sigma_j(t)]}, \\ 1 &= \prod_{t=0}^{t_m} \int dk(t) \delta[k(t) - k(t; \{\hat{\mathbf{h}}\})] \\ &= \left[\frac{N}{2\pi} \right]^{t_m+1} \int d\mathbf{k} d\hat{\mathbf{k}} e^{iN \sum_t \hat{k}(t)[k(t) - \frac{1}{N} \xi_j^1 \hat{h}_j(t)]}, \\ 1 &= \prod_{t, t'=0}^{t_m} \int dq(t, t') \delta[q(t, t') - q(t, t'; \{\sigma\})] \\ &= \left[\frac{N}{2\pi} \right]^{(t_m+1)^2} \int d\mathbf{q} d\hat{\mathbf{q}} e^{iN \sum_{t, t'} \hat{q}(t, t')[q(t, t') - \frac{1}{N} \sigma_j(t) \sigma_j(t')]}, \\ 1 &= \prod_{t, t'=0}^{t_m} \int dQ(t, t') \delta[Q(t, t') - Q(t, t'; \{\hat{\mathbf{h}}\})] \\ &= \left[\frac{N}{2\pi} \right]^{(t_m+1)^2} \int d\mathbf{Q} d\hat{\mathbf{Q}} e^{iN \sum_{t, t'} \hat{Q}(t, t')[Q(t, t') - \frac{1}{N} \hat{h}_j(t) \hat{h}_j(t')]}, \\ 1 &= \prod_{t, t'=0}^{t_m} \int dK(t, t') \delta[K(t, t') - K(t, t'; \{\sigma, \hat{\mathbf{h}}\})] \\ &= \left[\frac{N}{2\pi} \right]^{(t_m+1)^2} \int d\mathbf{K} d\hat{\mathbf{K}} e^{iN \sum_{t, t'} \hat{K}(t, t')[K(t, t') - \frac{1}{N} \hat{h}_j(t) \sigma_j(t')]}. \end{aligned}$$

Using the short-hand notation,

$$\begin{aligned} \Psi[\mathbf{a}, \hat{\mathbf{a}}, \mathbf{k}, \hat{\mathbf{k}}, \mathbf{q}, \hat{\mathbf{q}}, \mathbf{Q}, \hat{\mathbf{Q}}, \mathbf{K}, \hat{\mathbf{K}}] &= i \sum_t [\hat{a}(t)a(t) + \hat{k}(t)k(t)] \\ &+ i \sum_{t,t'} [\hat{q}(t,t')q(t,t') + \hat{Q}(t,t')Q(t,t') + \hat{K}(t,t')K(t,t')] \end{aligned}$$

then leads us to

$$\begin{aligned} \overline{Z[\psi]} &= \int d\mathbf{a}d\hat{\mathbf{a}}d\mathbf{k}d\hat{\mathbf{k}}d\mathbf{q}d\hat{\mathbf{q}}d\mathbf{Q}d\hat{\mathbf{Q}}d\mathbf{K}d\hat{\mathbf{K}} \\ &\times e^{N\Psi[\mathbf{a}, \hat{\mathbf{a}}, \mathbf{k}, \hat{\mathbf{k}}, \mathbf{q}, \hat{\mathbf{q}}, \mathbf{Q}, \hat{\mathbf{Q}}, \mathbf{K}, \hat{\mathbf{K}}] + N\Phi[\mathbf{a}, \mathbf{k}, \mathbf{q}, \mathbf{Q}, \mathbf{K}] + \mathcal{O}(\dots)} \\ &\times \int \{d\mathbf{h}d\hat{\mathbf{h}}\} \sum_{\boldsymbol{\sigma}(0)} \cdots \sum_{\boldsymbol{\sigma}(t_m)} p(\boldsymbol{\sigma}(0)) \\ &\times \prod_{it} e^{i\hat{h}_i(t)[h_i(t) - \theta_i(t)] - i\psi_i(t)\sigma_i(t) + \beta\sigma_i(t+1)h_i(t) - \log 2 \cosh[\beta h_i(t)]} \\ &\times \prod_i e^{-i\xi_i^1 \sum_t [\hat{a}(t)\sigma_i(t) + \hat{k}(t)\hat{h}_i(t)]} \\ &\quad - i \sum_{t,t'} [\hat{q}(t,t')\sigma_i(t)\sigma_i(t') + \hat{Q}(t,t')\hat{h}_i(t)\hat{h}_i(t') + \hat{K}(t,t')\hat{h}_i(t)\sigma_i(t')], \end{aligned}$$

in which the term denoted as $\mathcal{O}(\dots)$ covers both the non-dominant orders in (3.423) and the $\mathcal{O}(\log N)$ relics of the various pre-factors $[N/2\pi]$ in the above integral representations of the δ -distributions (note: t_m was assumed fixed). We now see explicitly that the summations and integrations over neuron states and local fields fully factorize over the N sites. A simple transformation

$$\{\sigma_i(t), h_i(t), \hat{h}_i(t)\} \rightarrow \{\xi_i^1 \sigma_i(t), \xi_i^1 h_i(t), \xi_i^1 \hat{h}_i(t)\}$$

brings the result into the form [Coo01, SC00, SC01, CKS05]

$$\begin{aligned} e^N \Xi[\hat{\mathbf{a}}, \hat{\mathbf{k}}, \hat{\mathbf{q}}, \hat{\mathbf{Q}}, \hat{\mathbf{K}}] &= \int \{d\mathbf{h}d\hat{\mathbf{h}}\} \sum_{\boldsymbol{\sigma}(0)} \cdots \sum_{\boldsymbol{\sigma}(t_m)} p(\boldsymbol{\sigma}(0)) \\ &\times \prod_{it} e^{i\hat{h}_i(t)[h_i(t) - \xi_i^1 \theta_i(t)] - i\xi_i^1 \psi_i(t)\sigma_i(t)} \\ &\quad + \beta\sigma_i(t+1)h_i(t) - \log 2 \cosh[\beta h_i(t)] \prod_i e^{-i\xi_i^1 \sum_t [\hat{a}(t)\sigma_i(t) + \hat{k}(t)\hat{h}_i(t)]} \\ &\quad - i \sum_{t,t'} [\hat{q}(t,t')\sigma_i(t)\sigma_i(t') + \hat{Q}(t,t')\hat{h}_i(t)\hat{h}_i(t') + \hat{K}(t,t')\hat{h}_i(t)\sigma_i(t')], \end{aligned}$$

in which

$$\{dh d\hat{h}\} = \prod_t [dh(t)d\hat{h}(t)/2\pi], \quad \pi_0(\sigma) = \frac{1}{2}[1 + m_0]\delta_{\sigma,1} + \frac{1}{2}[1 - m_0]\delta_{\sigma,-1}.$$

At this stage $\overline{Z[\psi]}$ acquires the form of an integral to be evaluated via the saddle-point (or ‘steepest descent’) method,

$$\overline{Z[\{\psi(t)\}]} = \int d\mathbf{a}d\hat{\mathbf{a}}d\mathbf{k}d\hat{\mathbf{k}}d\mathbf{q}d\hat{\mathbf{q}}d\mathbf{Q}d\hat{\mathbf{Q}}d\mathbf{K}d\hat{\mathbf{K}} e^{N\{\Psi[\dots] + \Phi[\dots] + \Xi[\dots]\} + \mathcal{O}(\dots)}. \quad (3.424)$$

The disorder-averaged partition function (3.424) is for $N \rightarrow \infty$ dominated by the physical saddle-point of the macroscopic surface

$$\Psi[\mathbf{a}, \hat{\mathbf{a}}, \mathbf{k}, \hat{\mathbf{k}}, \mathbf{q}, \hat{\mathbf{q}}, \mathbf{Q}, \hat{\mathbf{Q}}, \mathbf{K}, \hat{\mathbf{K}}] + \Phi[\mathbf{a}, \mathbf{k}, \mathbf{q}, \mathbf{Q}, \mathbf{K}] + \Xi[\hat{\mathbf{a}}, \hat{\mathbf{k}}, \hat{\mathbf{q}}, \hat{\mathbf{Q}}, \hat{\mathbf{K}}]. \quad (3.425)$$

It will be advantageous at this stage to define the following effective measure:

$$\langle f[\{\sigma\}, \{h\}, \{\hat{h}\}] \rangle_{\star} = (3.426)$$

$$\frac{1}{N} \left\{ \frac{\int \{dhd\hat{h}\} \sum_{\sigma(0)\dots\sigma(t_m)} M_i[\{\sigma\}, \{h\}, \{\hat{h}\}] f[\{\sigma\}, \{h\}, \{\hat{h}\}]}{\int \{dhd\hat{h}\} \sum_{\sigma(0)\dots\sigma(t_m)} M_i[\{\sigma\}, \{h\}, \{\hat{h}\}]} \right\},$$

with

$$M_i[\{\sigma\}, \{h\}, \{\hat{h}\}] =$$

$$\pi_0(\sigma(0)) e^{\sum_t \{i\hat{h}(t)[h(t) - \xi_i^1 \theta_i(t)] - i\xi_i^1 \psi_i(t)\sigma(t) + \beta\sigma(t+1)h(t) - \log 2 \cosh[\beta h(t)]\}}$$

$$\times e^{-i \sum_t [\hat{a}(t)\sigma(t) + \hat{k}(t)\hat{h}(t)] - i \sum_{t,t'} [\hat{q}(t,t')\sigma(t)\sigma(t') + \hat{Q}(t,t')\hat{h}(t)\hat{h}(t') + \hat{K}(t,t')\hat{h}(t)\sigma(t')]},$$

in which the values to be inserted for $\{\hat{m}(t), \hat{k}(t), \hat{q}(t, t'), \hat{Q}(t, t'), \hat{K}(t, t')\}$ are given by the saddle-point of (3.425). Variation of (3.425) with respect to all the original macroscopic objects occurring as arguments (those without the ‘hats’) gives the following set of saddle-point equations:

$$\hat{a}(t) = i \frac{\partial \Phi}{\partial a(t)}, \quad \hat{k}(t) = i \frac{\partial \Phi}{\partial k(t)}, \quad \hat{q}(t, t') = i \frac{\partial \Phi}{\partial q(t, t')},$$

$$\hat{Q}(t, t') = i \frac{\partial \Phi}{\partial Q(t, t')}, \quad \hat{K}(t, t') = i \frac{\partial \Phi}{\partial K(t, t')}.$$

Variation of (3.425) with respect to the conjugate macroscopic objects (those with the ‘hats’), in turn, and usage of our newly introduced short-hand notation $\langle \dots \rangle_{\star}$, gives:

$$a(t) = \langle \sigma(t) \rangle_{\star}, \quad k(t) = \langle \hat{h}(t) \rangle_{\star}, \quad q(t, t') = \langle \sigma(t)\sigma(t') \rangle_{\star},$$

$$Q(t, t') = \langle \hat{h}(t)\hat{h}(t') \rangle_{\star}, \quad K(t, t') = \langle \hat{h}(t)\sigma(t') \rangle_{\star}$$

The above coupled equations have to be solved simultaneously, once we have calculated the term $\Phi[\dots]$ that depends on the synapses. This appears to be a formidable task; it can, however, be simplified considerably upon first deriving the physical meaning of the above macroscopic quantities. We use identities such as

$$\frac{\partial \Xi[\dots]}{\partial \psi_j(t)} = -\frac{i}{N} \xi_j^1 \left[\frac{\int \{dhd\hat{h}\} \sum_{\sigma(0)\dots\sigma(t_m)} M_j[\{\sigma\}, \{h\}, \{\hat{h}\}]\sigma(t)}{\int \{dhd\hat{h}\} \sum_{\sigma(0)\dots\sigma(t_m)} M_j[\{\sigma\}, \{h\}, \{\hat{h}\}]} \right],$$

$$\frac{\partial \Xi[\dots]}{\partial \theta_j(t)} = -\frac{i}{N} \xi_j^1 \left[\frac{\int \{dhd\hat{h}\} \sum_{\sigma(0)\dots\sigma(t_m)} M_j[\{\sigma\}, \{h\}, \{\hat{h}\}]\hat{h}(t)}{\int \{dhd\hat{h}\} \sum_{\sigma(0)\dots\sigma(t_m)} M_j[\{\sigma\}, \{h\}, \{\hat{h}\}]} \right],$$

$$\begin{aligned} \frac{\partial^2 \Xi[\dots]}{\partial \psi_j(t) \partial \psi_j(t')} &= -\frac{1}{N} \left[\frac{\int \{dhd\hat{h}\} \sum_{\sigma(0)\dots\sigma(t_m)} M_j[\{\sigma\}, \{h\}, \{\hat{h}\}] \sigma(t) \sigma(t')}{\int \{dhd\hat{h}\} \sum_{\sigma(0)\dots\sigma(t_m)} M_j[\{\sigma\}, \{h\}, \{\hat{h}\}]} \right] \\ &\quad - N \left[\frac{\partial \Xi[\dots]}{\partial \psi_j(t)} \right] \left[\frac{\partial \Xi[\dots]}{\partial \psi_j(t')} \right], \end{aligned}$$

$$\begin{aligned} \frac{\partial^2 \Xi[\dots]}{\partial \theta_j(t) \partial \theta_j(t')} &= -\frac{1}{N} \left[\frac{\int \{dhd\hat{h}\} \sum_{\sigma(0)\dots\sigma(t_m)} M_j[\{\sigma\}, \{h\}, \{\hat{h}\}] \hat{h}(t) \hat{h}(t')}{\int \{dhd\hat{h}\} \sum_{\sigma(0)\dots\sigma(t_m)} M_j[\{\sigma\}, \{h\}, \{\hat{h}\}]} \right] \\ &\quad - N \left[\frac{\partial \Xi[\dots]}{\partial \theta_j(t)} \right] \left[\frac{\partial \Xi[\dots]}{\partial \theta_j(t')} \right], \end{aligned}$$

$$\begin{aligned} \frac{\partial^2 \Xi[\dots]}{\partial \psi_j(t) \partial \theta_j(t')} &= -\frac{i}{N} \left[\frac{\int \{dhd\hat{h}\} \sum_{\sigma(0)\dots\sigma(t_m)} M_j[\{\sigma\}, \{h\}, \{\hat{h}\}] \sigma(t) \hat{h}(t')}{\int \{dhd\hat{h}\} \sum_{\sigma(0)\dots\sigma(t_m)} M_j[\{\sigma\}, \{h\}, \{\hat{h}\}]} \right] \\ &\quad - N \left[\frac{\partial \Xi[\dots]}{\partial \psi_j(t)} \right] \left[\frac{\partial \Xi[\dots]}{\partial \theta_j(t')} \right], \end{aligned}$$

and using the short-hand notation (3.426) wherever possible. Note that the external fields $\{\psi_i(t), \theta_i(t)\}$ occur only in the function $\Xi[\dots]$, not in $\Psi[\dots]$ or $\Phi[\dots]$, and that overall constants in $\overline{Z[\psi]}$ can always be recovered *a posteriori*, using $\overline{Z[\mathbf{0}]} = 1$:

$$\begin{aligned} m(t) &= \lim_{\psi \rightarrow \mathbf{0}} \frac{i}{N} \sum_i \xi_i^1 \frac{\int d\mathbf{a} \dots d\hat{\mathbf{K}} \left[\frac{N \partial \Xi}{\partial \psi_i(t)} \right] e^{N[\Psi + \Phi + \Xi] + \mathcal{O}(\dots)}}{\int d\mathbf{a} \dots d\hat{\mathbf{K}} e^{N[\Psi + \Phi + \Xi] + \mathcal{O}(\dots)}} \\ &= \lim_{\psi \rightarrow \mathbf{0}} \langle \sigma(t) \rangle_{\star}, \\ C(t, t') &= -\lim_{\psi \rightarrow \mathbf{0}} \frac{1}{N} \sum_i \frac{\int d\mathbf{a} \dots d\hat{\mathbf{K}} \left[\frac{N \partial^2 \Xi}{\partial \psi_i(t) \partial \psi_i(t')} + \frac{N \partial \Xi}{\partial \psi_i(t)} \frac{N \partial \Xi}{\partial \psi_i(t')} \right] e^{N[\Psi + \Phi + \Xi] + \mathcal{O}(\dots)}}{\int d\mathbf{a} \dots d\hat{\mathbf{K}} e^{N[\Psi + \Phi + \Xi] + \mathcal{O}(\dots)}} \\ &= \lim_{\psi \rightarrow \mathbf{0}} \langle \sigma(t) \sigma(t') \rangle_{\star}, \\ iG(t, t') &= -\lim_{\psi \rightarrow \mathbf{0}} \frac{1}{N} \sum_i \frac{\int d\mathbf{a} \dots d\hat{\mathbf{K}} \left[\frac{N \partial^2 \Xi}{\partial \psi_i(t) \partial \theta_i(t')} + \frac{N \partial \Xi}{\partial \psi_i(t)} \frac{N \partial \Xi}{\partial \theta_i(t')} \right] e^{N[\Psi + \Phi + \Xi] + \mathcal{O}(\dots)}}{\int d\mathbf{a} \dots d\hat{\mathbf{K}} e^{N[\Psi + \Phi + \Xi] + \mathcal{O}(\dots)}} \\ &= \lim_{\psi \rightarrow \mathbf{0}} \langle \sigma(t) \hat{h}(t') \rangle_{\star}. \end{aligned}$$

Finally we get useful identities from the seemingly trivial statements $N^{-1} \sum_i \xi_i^1 \partial \overline{Z[\mathbf{0}]} / \partial \theta_i(t) = 0$ and $N^{-1} \sum_i \partial^2 \overline{Z[\mathbf{0}]} / \partial \theta_i(t) \partial \theta_i(t') = 0$,

$$\begin{aligned}
0 &= \lim_{\psi \rightarrow 0} \frac{i}{N} \sum_i \xi_i^1 \frac{\int d\mathbf{a} \dots d\hat{\mathbf{K}} \left[\frac{N\partial\Xi}{\partial\theta_i(t)} \right] e^{N[\Psi+\Phi+\Xi]+\mathcal{O}(\dots)}}{\int d\mathbf{a} \dots d\hat{\mathbf{K}} e^{N[\Psi+\Phi+\Xi]+\mathcal{O}(\dots)}} = \lim_{\psi \rightarrow 0} \langle \hat{h}(t) \rangle_\star, \\
0 &= - \lim_{\psi \rightarrow 0} \frac{1}{N} \sum_i \frac{\int d\mathbf{a} \dots d\hat{\mathbf{K}} \left[\frac{N\partial^2\Xi}{\partial\theta_i(t)\partial\theta_i(t')} + \frac{N\partial\Xi}{\partial\theta_i(t)} \frac{N\partial\Xi}{\partial\theta_i(t')} \right] e^{N[\Psi+\Phi+\Xi]+\mathcal{O}(\dots)}}{\int d\mathbf{a} \dots d\hat{\mathbf{K}} e^{N[\Psi+\Phi+\Xi]+\mathcal{O}(\dots)}} \\
&= \lim_{\psi \rightarrow 0} \langle \hat{h}(t)\hat{h}(t') \rangle_\star.
\end{aligned}$$

The above identities simplify our problem considerably. The dummy fields $\psi_i(t)$ have served their purpose and will now be put to zero, as a result we can now identify our macroscopic observables *at the relevant saddle-point* as:

$$\begin{aligned}
a(t) &= m(t), & k(t) &= 0, & q(t, t') &= C(t, t'), \\
Q(t, t') &= 0, & K(t, t') &= iG(t', t).
\end{aligned}$$

Finally we make a convenient choice for the external fields, $\theta_i(t) = \xi_i^1 \theta(t)$, with which the effective measure $\langle \dots \rangle_\star$ simplifies to

$$\langle f[\{\sigma\}, \{h\}, \{\hat{h}\}] \rangle_\star = \frac{\int \{dh d\hat{h}\} \sum_{\sigma(0) \dots \sigma(t_m)} M[\{\sigma\}, \{h\}, \{\hat{h}\}] f[\{\sigma\}, \{h\}, \{\hat{h}\}]}{\int \{dh d\hat{h}\} \sum_{\sigma(0) \dots \sigma(t_m)} M[\{\sigma\}, \{h\}, \{\hat{h}\}]}, \quad (3.427)$$

$$\begin{aligned}
&\text{with} \quad M[\{\sigma\}, \{h\}, \{\hat{h}\}] = \\
&\pi_0(\sigma(0)) e^{\sum_i \{i\hat{h}(t)[h(t)-\theta(t)] + \beta\sigma(t+1)h(t) - \log 2 \cosh[\beta h(t)]\} - i \sum_t [\hat{a}(t)\sigma(t) + \hat{k}(t)\hat{h}(t)]} \\
&\quad \times e^{-i \sum_{t,t'} [\hat{q}(t,t')\sigma(t)\sigma(t') + \hat{Q}(t,t')\hat{h}(t)\hat{h}(t') + \hat{K}(t,t')\hat{h}(t)\sigma(t')]} .
\end{aligned}$$

Our final task is calculating the leading order of

$$\mathcal{F}[\{\sigma\}, \{\hat{h}\}] = \frac{1}{N} \log \left[e^{-i \sum_{it} \hat{h}_i(t) J_{ij} \sigma_j(t)} \right]. \quad (3.428)$$

Parallel Hopfield Model Near Saturation

The fully connected Hopfield [Hop82] network (here with parallel dynamics) is got upon choosing $c = N$ in the recipe (3.421), i.e., $c_{ij} = 1 - \delta_{ij}$ and $p = \alpha N$. The disorder average thus involves only the patterns with $\mu > 1$. Now (3.428) gives

$$\begin{aligned}
\mathcal{F}[\dots] &= \frac{1}{N} \log \left[e^{-i N^{-1} \sum_t \xi_i^\mu \xi_j^\mu \hat{h}_i(t) \sigma_j(t)} \right] \\
&= i\alpha \sum_t K(t, t; \{\sigma, \hat{h}\}) - i \sum_t a(t) k(t) + \\
&\quad \alpha \log \left[e^{-i \sum_t [\xi_i \hat{h}_i(t)/\sqrt{N}] [\xi_i \sigma_i(t)/\sqrt{N}]} \right] + \mathcal{O}(N^{-1}).
\end{aligned} \quad (3.429)$$

We concentrate on the last term:

$$\begin{aligned}
\left[e^{-i \sum_t [\xi_i \hat{h}_i(t)/\sqrt{N}] [\xi_i \sigma_i(t)/\sqrt{N}]} \right] &= \int d\mathbf{x} d\mathbf{y} e^{-i\mathbf{x}\cdot\mathbf{y}} \\
&\times \prod_t \left\{ \delta\left[x(t) - \frac{\xi_i \sigma_i(t)}{\sqrt{N}}\right] \delta\left[y(t) - \frac{\xi_i \hat{h}_i(t)}{\sqrt{N}}\right] \right\} \\
&= \int \frac{d\mathbf{x} d\mathbf{y} d\hat{\mathbf{x}} d\hat{\mathbf{y}}}{(2\pi)^{2(t_m+1)}} e^{i[\hat{\mathbf{x}}\cdot\mathbf{x} + \hat{\mathbf{y}}\cdot\mathbf{y} - \mathbf{x}\cdot\mathbf{y}]} \\
&\times \overline{\left[e^{-\frac{i}{\sqrt{N}} \xi_i \sum_t [\hat{x}(t) \sigma_i(t) + \hat{y}(t) \hat{h}_i(t)]} \right]} \\
&= \int \frac{d\mathbf{x} d\mathbf{y} d\hat{\mathbf{x}} d\hat{\mathbf{y}}}{(2\pi)^{2(t_m+1)}} e^{i[\hat{\mathbf{x}}\cdot\mathbf{x} + \hat{\mathbf{y}}\cdot\mathbf{y} - \mathbf{x}\cdot\mathbf{y}]} \\
&\quad + \sum_i \log \cos \left[\frac{1}{\sqrt{N}} \sum_t [\hat{x}(t) \sigma_i(t) + \hat{y}(t) \hat{h}_i(t)] \right] \\
&= \int \frac{d\mathbf{x} d\mathbf{y} d\hat{\mathbf{x}} d\hat{\mathbf{y}}}{(2\pi)^{2(t_m+1)}} e^{i[\hat{\mathbf{x}}\cdot\mathbf{x} + \hat{\mathbf{y}}\cdot\mathbf{y} - \mathbf{x}\cdot\mathbf{y}]} \\
&\quad - \frac{1}{2N} \sum_i \{ \sum_t [\hat{x}(t) \sigma_i(t) + \hat{y}(t) \hat{h}_i(t)] \}^2 + \mathcal{O}(N^{-1}) \\
&= \int \frac{d\mathbf{x} d\mathbf{y} d\hat{\mathbf{x}} d\hat{\mathbf{y}}}{(2\pi)^{2(t_m+1)}} e^{i[\hat{\mathbf{x}}\cdot\mathbf{x} + \hat{\mathbf{y}}\cdot\mathbf{y} - \mathbf{x}\cdot\mathbf{y}]} \\
&\quad - \frac{1}{2} \sum_{t,t'} [\hat{x}(t) \hat{x}(t') q(t,t') + 2\hat{x}(t) \hat{y}(t') K(t',t) \\
&\quad + \hat{y}(t) \hat{y}(t') Q(t,t')].
\end{aligned}$$

Together with (3.429) we have now shown that the disorder average (3.428) is indeed, in leading order in N , with

$$\begin{aligned}
\Phi[\mathbf{a}, \mathbf{k}, \mathbf{q}, \mathbf{Q}, \mathbf{K}] &= i\alpha \sum_t K(t, t) - i\mathbf{a} \cdot \mathbf{k} + \alpha \log \int \frac{d\mathbf{x} d\mathbf{y} d\hat{\mathbf{x}} d\hat{\mathbf{y}}}{(2\pi)^{2(t_m+1)}} e^{i[\hat{\mathbf{x}}\cdot\mathbf{x} + \hat{\mathbf{y}}\cdot\mathbf{y} - \mathbf{x}\cdot\mathbf{y}]} \\
&\quad - \frac{1}{2} [\hat{\mathbf{x}} \cdot \mathbf{q} \hat{\mathbf{x}} + 2\hat{\mathbf{y}} \cdot \mathbf{K} \hat{\mathbf{x}} + \hat{\mathbf{y}} \cdot \mathbf{Q} \hat{\mathbf{y}}] = i\alpha \sum_t K(t, t) - i\mathbf{a} \cdot \mathbf{k} \\
&\quad + \alpha \log \int \frac{d\mathbf{u} d\mathbf{v}}{(2\pi)^{t_m+1}} e^{-\frac{1}{2} [\mathbf{u} \cdot \mathbf{q} \mathbf{u} + 2\mathbf{v} \cdot \mathbf{K} \mathbf{u} - 2i\mathbf{u} \cdot \mathbf{v} + \mathbf{v} \cdot \mathbf{Q} \mathbf{v}]}.
\end{aligned}$$

Now, for the single-time observables, this gives $\hat{a}(t) = k(t)$ and $\hat{k}(t) = a(t)$, and for the two-time ones:

$$\begin{aligned}
\hat{q}(t, t') &= -\frac{1}{2} \alpha i \frac{\int d\mathbf{u} d\mathbf{v} u(t) u(t') e^{-\frac{1}{2} [\mathbf{u} \cdot \mathbf{q} \mathbf{u} + 2\mathbf{v} \cdot \mathbf{K} \mathbf{u} - 2i\mathbf{u} \cdot \mathbf{v} + \mathbf{v} \cdot \mathbf{Q} \mathbf{v}]} }{\int d\mathbf{u} d\mathbf{v} e^{-\frac{1}{2} [\mathbf{u} \cdot \mathbf{q} \mathbf{u} + 2\mathbf{v} \cdot \mathbf{K} \mathbf{u} - 2i\mathbf{u} \cdot \mathbf{v} + \mathbf{v} \cdot \mathbf{Q} \mathbf{v}]}}, \\
\hat{Q}(t, t') &= -\frac{1}{2} \alpha i \frac{\int d\mathbf{u} d\mathbf{v} v(t) v(t') e^{-\frac{1}{2} [\mathbf{u} \cdot \mathbf{q} \mathbf{u} + 2\mathbf{v} \cdot \mathbf{K} \mathbf{u} - 2i\mathbf{u} \cdot \mathbf{v} + \mathbf{v} \cdot \mathbf{Q} \mathbf{v}]} }{\int d\mathbf{u} d\mathbf{v} e^{-\frac{1}{2} [\mathbf{u} \cdot \mathbf{q} \mathbf{u} + 2\mathbf{v} \cdot \mathbf{K} \mathbf{u} - 2i\mathbf{u} \cdot \mathbf{v} + \mathbf{v} \cdot \mathbf{Q} \mathbf{v}]}}, \\
\hat{K}(t, t') &= -\alpha i \frac{\int d\mathbf{u} d\mathbf{v} v(t) u(t') e^{-\frac{1}{2} [\mathbf{u} \cdot \mathbf{q} \mathbf{u} + 2\mathbf{v} \cdot \mathbf{K} \mathbf{u} - 2i\mathbf{u} \cdot \mathbf{v} + \mathbf{v} \cdot \mathbf{Q} \mathbf{v}]} }{\int d\mathbf{u} d\mathbf{v} e^{-\frac{1}{2} [\mathbf{u} \cdot \mathbf{q} \mathbf{u} + 2\mathbf{v} \cdot \mathbf{K} \mathbf{u} - 2i\mathbf{u} \cdot \mathbf{v} + \mathbf{v} \cdot \mathbf{Q} \mathbf{v}]} - \alpha \delta_{t,t'}.
\end{aligned}$$

At the physical saddle-point we can now express all non-zero objects in terms of the observables $m(t)$, $C(t, t')$ and $G(t, t')$, with a clear physical meaning. Thus we find $\hat{a}(t) = 0$, $\hat{k}(t) = m(t)$, and

$$\begin{aligned}
\hat{q}(t, t') &= -\frac{1}{2}\alpha i \frac{\int d\mathbf{u}d\mathbf{v} u(t)u(t')e^{-\frac{1}{2}[\mathbf{u}\cdot\mathbf{C}\mathbf{u}-2i\mathbf{u}\cdot[\mathbf{1}-\mathbf{G}]\mathbf{v}]} }{\int d\mathbf{u}d\mathbf{v} e^{-\frac{1}{2}[\mathbf{u}\cdot\mathbf{C}\mathbf{u}-2i\mathbf{u}\cdot[\mathbf{1}-\mathbf{G}]\mathbf{v}]} } = 0, \\
\hat{Q}(t, t') &= -\frac{1}{2}\alpha i \frac{\int d\mathbf{u}d\mathbf{v} v(t)v(t')e^{-\frac{1}{2}[\mathbf{u}\cdot\mathbf{C}\mathbf{u}-2i\mathbf{u}\cdot[\mathbf{1}-\mathbf{G}]\mathbf{v}]} }{\int d\mathbf{u}d\mathbf{v} e^{-\frac{1}{2}[\mathbf{u}\cdot\mathbf{C}\mathbf{u}-2i\mathbf{u}\cdot[\mathbf{1}-\mathbf{G}]\mathbf{v}]} } \\
&= -\frac{1}{2}\alpha i [(\mathbf{1}-\mathbf{G})^{-1}\mathbf{C}(\mathbf{1}-\mathbf{G}^\dagger)^{-1}] (t, t'), \\
\hat{K}(t, t') + \alpha\delta_{t,t'} &= -\alpha i \frac{\int d\mathbf{u}d\mathbf{v} v(t)u(t')e^{-\frac{1}{2}[\mathbf{u}\cdot\mathbf{C}\mathbf{u}-2i\mathbf{u}\cdot[\mathbf{1}-\mathbf{G}]\mathbf{v}]} }{\int d\mathbf{u}d\mathbf{v} e^{-\frac{1}{2}[\mathbf{u}\cdot\mathbf{C}\mathbf{u}-2i\mathbf{u}\cdot[\mathbf{1}-\mathbf{G}]\mathbf{v}]} } \\
&= \alpha(\mathbf{1}-\mathbf{G})^{-1}(t, t'),
\end{aligned}$$

with $G^\dagger(t, t') = G(t', t)$, and using standard manipulations of Gaussian integrals. Note that we can use the identity $(\mathbf{1}-\mathbf{G})^{-1} - \mathbf{1} = \sum_{\ell \geq 0} \mathbf{G}^\ell - \mathbf{1} = \sum_{\ell \geq 0} \mathbf{G}^\ell = \mathbf{G}(\mathbf{1}-\mathbf{G})^{-1}$ to compactify the last equation to

$$\hat{K}(t, t') = \alpha[\mathbf{G}(\mathbf{1}-\mathbf{G})^{-1}](t, t'). \quad (3.430)$$

We have now expressed all our objects in terms of the disorder-averaged recall Hopfield overlap $\mathbf{m} = \{m(t)\}$ and the disorder-averaged single-site correlation- and response-functions $\mathbf{C} = \{C(t, t')\}$ and $\mathbf{G} = \{G(t, t')\}$. We can next simplify the effective measure (3.427), which plays a crucial role in the remaining saddle-point equations. Inserting $\hat{a}(t) = \hat{q}(t, t') = 0$ and $\hat{k}(t) = m(t)$ into (3.427), first of all, gives us

$$M[\{\sigma\}, \{h\}, \{\hat{h}\}] = \pi_0(\sigma(0))$$

$$\begin{aligned}
&\times e^{\sum_t \left\{ i\hat{h}(t)[h(t)-m(t)-\theta(t)-\sum_{t'} \hat{K}(t, t')\sigma(t')] + \beta\sigma(t+1)h(t) - \log 2 \cosh[\beta h(t)] \right\} - i \sum_{t, t'} \hat{Q}(t, t')\hat{h}(t)\hat{h}(t')}.
\end{aligned} \quad (3.431)$$

Secondly, causality ensures that $G(t, t') = 0$ for $t \leq t'$, from which, in combination with (3.430), it follows that the same must be true for the kernel $\hat{K}(t, t')$, since

$$\hat{K}(t, t') = \alpha[\mathbf{G}(\mathbf{1}-\mathbf{G})^{-1}](t, t') = \alpha\{\mathbf{G} + \mathbf{G}^2 + \mathbf{G}^3 + \dots\}(t, t').$$

This, in turn, guarantees that the function $M[\dots]$ in (3.431) is already normalized:

$$\int \{dh d\hat{h}\} \sum_{\sigma(0) \dots \sigma(t_m)} M[\{\sigma\}, \{h\}, \{\hat{h}\}] = 1.$$

One can prove this iteratively. After summation over $\sigma(t_m)$ (which due to causality cannot occur in the term with the kernel $\hat{K}(t, t')$) one is left with just a single occurrence of the field $h(t_m)$ in the exponent, integration over which reduces to $\delta[\hat{h}(t_m)]$, which then eliminates the conjugate field $\hat{h}(t_m)$. This cycle of operations is next applied to the variables at time $t_m - 1$, etc. The effective measure (3.427) can now be written simply as

$$\langle f[\{\sigma\}, \{h\}, \{\hat{h}\}] \rangle_{\star} = \sum_{\sigma(0) \cdots \sigma(t_m)} \int \{dh d\hat{h}\} M[\{\sigma\}, \{h\}, \{\hat{h}\}] f[\{\sigma\}, \{h\}, \{\hat{h}\}],$$

with $M[\dots]$ as given in (3.431). The remaining saddle-point equations to be solved, which can be slightly simplified by using the identity

$$\begin{aligned} \langle \sigma(t) \hat{h}(t') \rangle_{\star} &= i \partial \langle \sigma(t) \rangle_{\star} / \partial \theta(t'), \quad \text{are} \\ m(t) &= \langle \sigma(t) \rangle_{\star}, \quad C(t, t') = \langle \sigma(t) \sigma(t') \rangle_{\star}, \quad G(t, t') = \partial \langle \sigma(t) \rangle_{\star} / \partial \theta(t'). \end{aligned}$$

Here we observe that we only need to insert functions of spin states into the effective measure $\langle \dots \rangle_{\star}$ (rather than fields or conjugate fields), so the effective measure can again be simplified. We get

$$\langle f[\{\sigma\}] \rangle_{\star} = \sum_{\sigma(0) \cdots \sigma(t_m)} \text{Prob}[\{\sigma\}] f[\{\sigma\}], \quad \text{with}$$

$$\begin{aligned} \text{Prob}[\{\sigma\}] &= \pi_0(\sigma(0)) \int \{d\phi\} P[\{\phi\}] \\ &\times \prod_t \left[\frac{1}{2} [1 + \sigma(t+1) \tanh[\beta h(t|\{\sigma\}, \{\phi\})]] \right], \end{aligned} \quad (3.432)$$

in which $\pi_0(\sigma(0)) = \frac{1}{2} [1 + \sigma(0)m_0]$, and

$$\begin{aligned} h(t|\{\sigma\}, \{\phi\}) &= m(t) + \theta(t) + \alpha \sum_{t' < t} [\mathbf{G}(\mathbf{1} - \mathbf{G})^{-1}](t, t') \sigma(t') + \alpha^{\frac{1}{2}} \phi(t), \quad (3.433) \\ P[\{\phi\}] &= \frac{e^{-\frac{1}{2} \sum_{t, t'} \phi(t) [(\mathbf{1} - \mathbf{G}^{\dagger}) \mathbf{C}^{-1} (\mathbf{1} - \mathbf{G})](t, t') \phi(t')}}{(2\pi)^{(t_m+1)/2} \det^{-\frac{1}{2}} [(\mathbf{1} - \mathbf{G}^{\dagger}) \mathbf{C}^{-1} (\mathbf{1} - \mathbf{G})]}. \end{aligned}$$

We recognize (3.432) as describing an effective single neuron, with the usual dynamics $\text{Prob}[\sigma(t+1) = \pm 1] = \frac{1}{2} [1 \pm \tanh[\beta h(t)]]$, but with the fields (3.433). This result is indeed of the form (3.420), with a retarded self-interaction kernel $R(t, t')$ and covariance matrix $\langle \phi(t) \phi(t') \rangle$ of the Gaussian $\phi(t)$ given by

$$\begin{aligned} R(t, t') &= [\mathbf{G}(\mathbf{1} - \mathbf{G})^{-1}](t, t'), \\ \langle \phi(t) \phi(t') \rangle &= [(\mathbf{1} - \mathbf{G})^{-1} \mathbf{C} (\mathbf{1} - \mathbf{G}^{\dagger})^{-1}](t, t'). \end{aligned}$$

For $\alpha \rightarrow 0$ we lose all the complicated terms in the local fields, and recover the type of simple expression we found earlier for finite p : $m(t+1) = \tanh[\beta(m(t) + \theta(t))]$.

Note that always $C(t, t) = \langle \sigma^2(t) \rangle_{\star} = 1$ and $G(t, t') = R(t, t') = 0$ for $t \leq t'$. As a result the covariance matrix of the Gaussian fields can be written as

$$\begin{aligned}
\langle \phi(t)\phi(t') \rangle &= [(\mathbf{1} - \mathbf{G})^{-1} \mathbf{C} (\mathbf{1} - \mathbf{G}^\dagger)^{-1}] (t, t') \\
&= \sum_{s, s' \geq 0} [\delta_{t,s} + R(t, s)] C(s, s') [\delta_{s', t'} + R(t', s')] \\
&= \sum_{s=0}^t \sum_{s'=0}^{t'} [\delta_{t,s} + R(t, s)] C(s, s') [\delta_{s', t'} + R(t', s')].
\end{aligned}$$

Considering arbitrary positive integer powers of the response function immediately shows that

$$\begin{aligned}
(\mathbf{G}^\ell)(t, t') &= 0, \quad \text{if } t' > t - \ell, \quad \text{which gives} \\
R(t, t') &= \sum_{\ell > 0} (\mathbf{G}^\ell)(t, t') = \sum_{\ell=1}^{t-t'} (\mathbf{G}^\ell)(t, t').
\end{aligned}$$

Similarly we get from $(\mathbf{1} - \mathbf{G})^{-1} = \mathbf{1} + \mathbf{R}$ that for $t' \geq t$: $(\mathbf{1} - \mathbf{G})^{-1}(t, t') = \delta_{t, t'}$. To suppress notation we will simply put $h(t|..)$ instead of $h(t|\{\sigma\}, \{\phi\})$; this need not cause any ambiguity. We notice that summation over neuron variables $\sigma(s)$ and integration over Gaussian variables $\phi(s)$ with time arguments s higher than those occurring in the function to be averaged can always be carried out immediately, giving (for $t > 0$ and $t' < t$):

$$\begin{aligned}
m(t) &= \sum_{\sigma(0) \dots \sigma(t-1)} \pi_0(\sigma(0)) \int \{d\phi\} P[\{\phi\}] \tanh[\beta h(t-1|..)] \\
&\quad \times \prod_{s=0}^{t-2} \frac{1}{2} [1 + \sigma(s+1) \tanh[\beta h(s|..)]], \\
G(t, t') &= \beta \{ C(t, t' + 1) - \sum_{\sigma(0) \dots \sigma(t-1)} \pi_0(\sigma(0)) \\
&\quad \times \int \{d\phi\} P[\{\phi\}] \tanh[\beta h(t-1|..)] \tanh[\beta h(t'|..)] \\
&\quad \times \prod_{s=0}^{t-2} \frac{1}{2} [1 + \sigma(s+1) \tanh[\beta h(s|..)]],
\end{aligned}$$

which we get directly for $t' = t - 1$, and which follows for times $t' < t - 1$ upon using the identity

$$\sigma[1 - \tanh^2(x)] = [1 + \sigma \tanh(x)][\sigma - \tanh(x)].$$

For the correlations we distinguish between $t' = t - 1$ and $t' < t - 1$,

$$\begin{aligned}
C(t, t-1) &= \sum_{\sigma(0) \dots \sigma(t-2)} \pi_0(\sigma(0)) \int \{d\phi\} P[\{\phi\}] \tanh[\beta h(t-1|..)] \\
&\quad \times \tanh[\beta h(t-2|..)] \prod_{s=0}^{t-3} \frac{1}{2} [1 + \sigma(s+1) \tanh[\beta h(s|..)]],
\end{aligned}$$

whereas for $t' < t - 1$ we have

$$C(t, t') = \sum_{\sigma(0) \dots \sigma(t-1)} \pi_0(\sigma(0)) \int \{d\phi\} P[\{\phi\}] \tanh[\beta h(t-1|..)] \\ \times \sigma(t') \prod_{s=0}^{t-2} \frac{1}{2} [1 + \sigma(s+1) \tanh[\beta h(s|..)]] .$$

Now, the field at $t = 0$ is $h(0|..) = m_0 + \theta(0) + \alpha^{\frac{1}{2}}\phi(0)$, since the retarded self-interaction does not yet come into play. The distribution of $\phi(0)$ is fully characterized by its variance $\langle \phi^2(0) \rangle = C(0, 0) = 1$. Therefore, with $Dz = (2\pi)^{-\frac{1}{2}} e^{-\frac{1}{2}z^2} dz$, we immediately find

$$m(1) = \int Dz \tanh[\beta(m_0 + \theta(0) + z\sqrt{\alpha})], \quad C(1, 0) = m_0 m(1), \\ G(1, 0) = \beta \left\{ 1 - \int Dz \tanh^2[\beta(m_0 + \theta(0) + z\sqrt{\alpha})] \right\} .$$

For the self-interaction kernel this implies that $R(1, 0) = G(1, 0)$. We now move on to $t = 2$,

$$m(2) = \frac{1}{2} \sum_{\sigma(0)} \int d\phi(0) d\phi(1) P[\phi(0), \phi(1)] \tanh[\beta h(1|..)] [1 + \sigma(0)m_0],$$

$$C(2, 1) = \frac{1}{2} \sum_{\sigma(0)} \int d\phi(1) d\phi(0) P[\phi(0), \phi(1)] \\ \times \tanh[\beta h(1|..)] \tanh[\beta h(0|..)] [1 + \sigma(0)m_0]$$

$$C(2, 0) = \frac{1}{2} \sum_{\sigma(0)\sigma(1)} \int \{d\phi\} P[\{\phi\}] \tanh[\beta h(1|..)] \\ \times \sigma(0) \frac{1}{2} [1 + \sigma(1) \tanh[\beta h(0|..)]] [1 + \sigma(0)m_0],$$

$$G(2, 1) = \beta \left\{ 1 - \frac{1}{2} \sum_{\sigma(0)} \int d\phi(0) d\phi(1) P[\phi(0), \phi(1)] \right. \\ \left. \times \tanh^2[\beta h(1|..)] [1 + \sigma(0)m_0] \right\},$$

$$G(2, 0) = \beta \left\{ C(2, 1) - \frac{1}{2} \sum_{\sigma(0)} \int d\phi(0) d\phi(1) P[\phi(0), \phi(1)] \right. \\ \left. \times \tanh[\beta h(1|..)] \tanh[\beta h(0|..)] [1 + \sigma(0)m_0] \right\} = 0.$$

We already know that $\langle \phi^2(0) \rangle = 1$; the remaining two moments we need in order to determine $P[\phi(0), \phi(1)]$ read [Coo01, SC00, SC01, CKS05]

$$\begin{aligned}\langle \phi(1)\phi(0) \rangle &= \sum_{s=0}^1 [\delta_{1,s} + \delta_{0,s}R(1,0)]C(s,0) = C(1,0) + G(1,0), \\ \langle \phi^2(1) \rangle &= \sum_{s=0}^1 \sum_{s'=1}^1 [\delta_{1,s} + \delta_{0,s}R(1,0)]C(s,s')[\delta_{s',1} + \delta_{s',0}R(1,0)] \\ &= G^2(1,0) + 2C(0,1)G(1,0) + 1.\end{aligned}$$

We now know $P[\phi(0), \phi(1)]$ and can work out all macroscopic objects with $t = 2$ explicitly, if we wish. I will not do this here in full, but only point at the emerging pattern of all calculations at a given time t depending only on macroscopic quantities that have been calculated at times $t' < t$, which allows for iterative solution. Let us just work out $m(2)$ explicitly, in order to compare the first two recall overlaps $m(1)$ and $m(2)$ with the values found in simulations and in approximate theories. We note that calculating $m(2)$ only requires the field $\phi(1)$, for which we found $\langle \phi^2(1) \rangle = G^2(1,0) + 2C(0,1)G(1,0) + 1$:

$$\begin{aligned}m(2) &= \frac{1}{2} \sum_{\sigma(0)} \int d\phi(1) P[\phi(1)] \tanh[\beta(m(1) + \theta(1) \\ &\quad + \alpha G(1,0)\sigma(0) + \alpha^{\frac{1}{2}}\phi(1))][1 + \sigma(0)m_0] \\ &= \frac{1}{2} [1 + m_0] \int Dz \tanh[\beta(m(1) + \theta(1) + \alpha G(1,0) \\ &\quad + z\sqrt{\alpha[G^2(1,0) + 2m_0 m(1) G(1,0) + 1]})] \\ &\quad + \frac{1}{2} [1 - m_0] \int Dz \tanh[\beta(m(1) + \theta(1) - \alpha G(1,0) \\ &\quad + z\sqrt{\alpha[G^2(1,0) + 2m_0 m(1) G(1,0) + 1]})]\end{aligned}$$

Here we give a comparison of some of the approximate theories, the (exact) partition function (i.e., path-integral) formalism, and numerical simulations, for the case $\theta(t) = 0$ on the fully connected networks. The evolution of the recall overlap in the first two time-steps has been described as follows:

Naive Gaussian Approx: $m(1) = \int Dz \tanh[\beta(m(0) + z\sqrt{\alpha})]$

$$m(2) = \int Dz \tanh[\beta(m(1) + z\sqrt{\alpha})]$$

Amari-Maginu Theory: $m(1) = \int Dz \tanh[\beta(m(0) + z\sqrt{\alpha})]$

$$m(2) = \int Dz \tanh[\beta(m(1) + z\Sigma\sqrt{\alpha})]$$

$$\Sigma^2 = 1 + 2m(0)m(1)G + G^2$$

$$G = \beta \left[1 - \int Dz \tanh^2[\beta(m(0) + z\sqrt{\alpha})] \right]$$

Exact Solution:

$$m(1) = \int Dz \tanh[\beta(m(0) + z\sqrt{\alpha})]$$

$$m(2) = \frac{1}{2}[1 + m_0] \int Dz \tanh[\beta(m(1) + \alpha G + z\Sigma\sqrt{\alpha})]$$

$$+ \frac{1}{2}[1 - m_0] \int Dz \tanh[\beta(m(1) - \alpha G + z\Sigma\sqrt{\alpha})]$$

$$\Sigma^2 = 1 + 2m(0)m(1)G + G^2$$

$$G = \beta \left[1 - \int Dz \tanh^2[\beta(m(0) + z\sqrt{\alpha})] \right]$$

We can now appreciate why the more advanced Gaussian approximation (Amari–Maginu theory, [AM88]) works well when the system state is close to the target attractor. This theory gets the moments of the Gaussian part of the interference noise distribution at $t = 1$ exactly right, but not the discrete part, whereas close to the attractor both the response function $G(1, 0)$ and one of the two pre-factors $\frac{1}{2}[1 \pm m_0]$ in the exact expression for $m(2)$ will be very small, and the latter will therefore indeed approach a Gaussian shape. One can also see why the non-Gaussian approximation of [HO90] made sense: in the calculation of $m(2)$ the interference noise distribution can indeed be written as the sum of two Gaussian ones (although for $t > 2$ this will cease to be true).

Extremely Diluted Attractor Networks Near Saturation

The extremely diluted attractor networks were first studied in [DGZ87] (asymmetric dilution) and [WS91] (symmetric dilution). These models are got upon choosing $\lim_{N \rightarrow \infty} c/N = 0$ (while still $c \rightarrow \infty$) in definition (3.421) of the Hebbian synapses. The disorder average now involves both the patterns with $\mu > 1$ and the realization of the ‘wiring’ variables $c_{ij} \in \{0, 1\}$. Again, in working out the key function (3.428) we will show that for $N \rightarrow \infty$ the outcome can be written in terms of the above macroscopic quantities. We carry out the average over the spatial structure variables $\{c_{ij}\}$ first:

$$\begin{aligned} \mathcal{F}[\dots] &= \frac{1}{N} \log \left[e^{-\frac{i}{c} c_{ij} \xi_i^\mu \xi_j^\mu \sum_t \hat{h}_i(t) \sigma_j(t)} \right] \\ &= \frac{1}{N} \log \prod_{i < j} e^{-\frac{i}{c} \xi_i^\mu \xi_j^\mu [c_{ij} \sum_t \hat{h}_i(t) \sigma_j(t) + c_{ji} \sum_t \hat{h}_j(t) \sigma_i(t)]}. \end{aligned}$$

Now we have to distinguish between *symmetric* and *asymmetric dilution*. First we deal with the case of *symmetric dilution*: $c_{ij} = c_{ji}$ for all $i \neq j$. The average

over the c_{ij} is trivial:

$$\begin{aligned}
& \overline{\prod_{i < j} e^{-\frac{i}{c} c_{ij} \sum_{\mu} \xi_i^{\mu} \xi_j^{\mu} \sum_t [\hat{h}_i(t) \sigma_j(t) + \hat{h}_j(t) \sigma_i(t)]}} \\
&= \overline{\prod_{i < j} \left\{ 1 + \frac{c}{N} [e^{-\frac{i}{c} \xi_i^{\mu} \xi_j^{\mu} \sum_t [\hat{h}_i(t) \sigma_j(t) + \hat{h}_j(t) \sigma_i(t)]} - 1] \right\}} \\
&= \overline{\prod_{i < j} \left\{ 1 - \frac{c}{N} \left[\frac{i}{c} \xi_i^{\mu} \xi_j^{\mu} \sum_t [\hat{h}_i(t) \sigma_j(t) + \hat{h}_j(t) \sigma_i(t)] \right. \right. \\
&\quad \left. \left. + \frac{1}{2c^2} [\xi_i^{\mu} \xi_j^{\mu} \sum_t [\hat{h}_i(t) \sigma_j(t) + \hat{h}_j(t) \sigma_i(t)]]^2 + \mathcal{O}(c^{-\frac{3}{2}}) \right] \right\}} \\
&= \overline{\prod_{i < j} e^{-\frac{i}{N} \xi_i^{\mu} \xi_j^{\mu} \sum_t [\hat{h}_i(t) \sigma_j(t) + \hat{h}_j(t) \sigma_i(t)]}} \\
&\quad \overline{-\frac{1}{2cN} [\xi_i^{\mu} \xi_j^{\mu} \sum_t [\hat{h}_i(t) \sigma_j(t) + \hat{h}_j(t) \sigma_i(t)]]^2 + \mathcal{O}(\frac{1}{N\sqrt{c}}) + \mathcal{O}(\frac{c}{N^2})}.
\end{aligned}$$

We separate in the exponent the terms where $\mu = \nu$ in the quadratic term (being of the form $\sum_{\mu\nu} \dots$), and the terms with $\mu = 1$. We get (note: $p = \alpha c$):

$$\begin{aligned}
\mathcal{F}[\dots] &= -i \sum_t a(t) k(t) - \frac{1}{2} \alpha \sum_{st} [q(s, t) Q(s, t) + K(s, t) K(t, s)] \\
&\quad + \mathcal{O}(c^{-\frac{1}{2}}) + \mathcal{O}(c/N) \\
&\quad + \frac{1}{N} \log \left\{ \overline{e^{-\frac{i}{N} \sum_i [\xi_i^{\mu} \hat{h}_i(t)] [\xi_j^{\mu} \sigma_j(t)] - \frac{1}{4cN} \sum_{st} \xi_i^{\mu} \xi_j^{\mu} \xi_i^{\nu} \xi_j^{\nu} \right.} \\
&\quad \left. \overline{[\hat{h}_i(s) \sigma_j(s) + \hat{h}_j(s) \sigma_i(s)] [\hat{h}_i(t) \sigma_j(t) + \hat{h}_j(t) \sigma_i(t)]}} \right\}.
\end{aligned}$$

Our ‘condensed ansatz’ implies that for $\mu > 1$: $N^{-\frac{1}{2}} \xi_i^{\mu} \sigma_i(t) = \mathcal{O}(1)$ and $N^{-\frac{1}{2}} \xi_i^{\mu} \hat{h}_i(t) = \mathcal{O}(1)$. Thus the first term in the exponent containing the disorder is $\mathcal{O}(c)$, contributing $\mathcal{O}(c/N)$ to $\mathcal{F}[\dots]$. We therefore retain only the second term in the exponent. However, the same argument applies to the second term. There all contributions can be seen as uncorrelated in leading order, so that $\sum_{i \neq j} \sum_{\mu \neq \nu} \dots = \mathcal{O}(Np)$, giving a non-leading $\mathcal{O}(N^{-1})$ cumulative contribution to $\mathcal{F}[\dots]$. Thus, provided $\lim_{N \rightarrow \infty} c^{-1} = \lim_{N \rightarrow \infty} c/N = 0$ (which we assumed), we have shown that the disorder average (3.428) is again, in leading order in N , with

$$\begin{aligned}
\text{Symmetric Case: } \quad \Phi[\mathbf{a}, \mathbf{k}, \mathbf{q}, \mathbf{Q}, \mathbf{K}] &= -i \mathbf{a} \cdot \mathbf{k} - \frac{1}{2} \alpha \sum_{st} [q(s, t) Q(s, t) \\
&\quad + K(s, t) K(t, s)].
\end{aligned}$$

Next, we deal with the asymmetric case, where c_{ij} and c_{ji} are independent. Again, the average over the c_{ij} is trivial; here it gives

$$\begin{aligned}
& \overline{\prod_{i < j} \left\{ e^{-\frac{i}{c} c_{ij} \xi_i^\mu \xi_j^\mu \sum_t \hat{h}_i(t) \sigma_j(t)} e^{-\frac{i}{c} c_{ji} \xi_i^\mu \xi_j^\mu \sum_t \hat{h}_j(t) \sigma_i(t)} \right\}} \\
&= \overline{\prod_{i < j} \left\{ 1 + \frac{c}{N} [e^{-\frac{i}{c} \xi_i^\mu \xi_j^\mu \sum_t \hat{h}_i(t) \sigma_j(t)} - 1] \right\} \left\{ 1 + \frac{c}{N} [e^{-\frac{i}{c} \xi_i^\mu \xi_j^\mu \sum_t \hat{h}_j(t) \sigma_i(t)} - 1] \right\}} \\
&= \overline{\prod_{i < j} \left\{ 1 - \frac{c}{N} [\frac{i}{c} \xi_i^\mu \xi_j^\mu \sum_t \hat{h}_i(t) \sigma_j(t) + \frac{1}{2c^2} [\xi_i^\mu \xi_j^\mu \sum_t \hat{h}_i(t) \sigma_j(t)]^2 + \mathcal{O}(c^{-\frac{3}{2}})] \right\}} \\
&\quad \times \left\{ 1 - \frac{c}{N} [\frac{i}{c} \xi_i^\mu \xi_j^\mu \sum_t \hat{h}_j(t) \sigma_i(t) + \frac{1}{2c^2} [\xi_i^\mu \xi_j^\mu \sum_t \hat{h}_j(t) \sigma_i(t)]^2 + \mathcal{O}(c^{-\frac{3}{2}})] \right\} \\
&= \overline{\prod_{i < j} e^{-\frac{i}{N} \xi_i^\mu \xi_j^\mu \sum_t [\hat{h}_i(t) \sigma_j(t) + \hat{h}_j(t) \sigma_i(t)] - \frac{1}{2cN} [\xi_i^\mu \xi_j^\mu \sum_t \hat{h}_i(t) \sigma_j(t)]^2} \\
&\quad - \frac{1}{2cN} [\xi_i^\mu \xi_j^\mu \sum_t \hat{h}_j(t) \sigma_i(t)]^2 + \mathcal{O}(\frac{1}{N\sqrt{c}}) + \mathcal{O}(\frac{c}{N^2})}
\end{aligned}$$

Again, we separate in the exponent the terms where $\mu = \nu$ in the quadratic term and the terms with $\mu = 1$ and get

$$\begin{aligned}
\mathcal{F}[\dots] &= -i \sum_t a(t) k(t) - \frac{1}{2} \alpha \sum_{st} q(s, t) Q(s, t) + \mathcal{O}(c^{-\frac{1}{2}}) + \mathcal{O}(c/n) \\
&\quad + \frac{1}{N} \log \left\{ \overline{e^{-\frac{i}{N} \sum_t [\xi_i^\mu \hat{h}_i(t)] [\xi_j^\mu \sigma_j(t)] - \frac{1}{2cN} \xi_i^\mu \xi_j^\mu \xi_i^\nu \xi_j^\nu \sum_{st} \hat{h}_i(s) \sigma_j(s) \hat{h}_i(t) \sigma_j(t)}} \right\}.
\end{aligned}$$

The scaling arguments given in the symmetric case, based on our ‘condensed ansatz’, apply again, and tell us that the remaining terms with the disorder are of vanishing order in N . We have again shown that the disorder average (3.428) is, in leading order in N with

$$\text{Asymmetric Case:} \quad \Phi[\mathbf{a}, \mathbf{k}, \mathbf{q}, \mathbf{Q}, \mathbf{K}] = -i \mathbf{a} \cdot \mathbf{k} - \frac{1}{2} \alpha \sum_{st} q(s, t) Q(s, t).$$

Now, asymmetric dilution corresponds to $\Delta = 0$, i.e., there is no retarded self-interaction, and the response function no longer plays a role. We now only retain $h(t|\dots) = m(t) + \theta(t) + \alpha^{\frac{1}{2}} \phi(t)$, with $\langle \phi^2(t) \rangle = C(1, 1) = 1$. We now get

$$\begin{aligned}
m(t+1) &= \sum_{\sigma(0) \dots \sigma(t)} \pi_0(\sigma(0)) \int \{d\phi\} P[\{\phi\}] \tanh[\beta h(t|\dots)] \\
&\quad \times \prod_{s=0}^{t-1} \frac{1}{2} [1 + \sigma(s+1) \tanh[\beta h(s|\dots)]] \\
&= \int Dz \tanh[\beta(m(t) + \theta(t) + z\sqrt{\alpha})].
\end{aligned}$$

Similarly, for $t > t'$ equations for correlation and response functions reduce to

$$\begin{aligned}
C(t, t') &= \int \frac{d\phi_a d\phi_b}{2\pi\sqrt{1-C^2(t-1, t'-1)}} e^{-\frac{1}{2} \frac{\phi_a^2 + \phi_b^2 - 2C(t-1, t'-1)\phi_a\phi_b}{1-C^2(t-1, t'-1)}} \\
&\quad \times \tanh[\beta(m(t-1) + \theta(t-1) + \phi_a\sqrt{\alpha})] \\
&\quad \times \tanh[\beta(m(t'-1) + \theta(t'-1) + \phi_b\sqrt{\alpha})], \\
G(t, t') &= \beta\delta_{t, t'+1} \left\{ 1 - \int Dz \tanh^2[\beta(m(t-1) + \theta(t-1) + z\sqrt{\alpha})] \right\}.
\end{aligned}$$

Let us also inspect the stationary state $m(t) = m$, for $\theta(t) = 0$. One easily proves that $m = 0$ as soon as $T > 1$, using

$$m^2 = \beta m \int_0^m dk [1 - \int Dz \tanh^2[\beta(k + z\sqrt{\alpha})]] \leq \beta m^2.$$

A continuous bifurcation occurs from the $m = 0$ state to an $m > 0$ state when $T = 1 - \int Dz \tanh^2[\beta z\sqrt{\alpha}]$. A parametrization of this transition line in the (α, T) -plane is given by

$$T(x) = 1 - \int Dz \tanh^2(zx), \quad \alpha(x) = x^2 T^2(x), \quad \text{for } x \geq 0.$$

For $\alpha = 0$ we just get $m = \tanh(\beta m)$ so $T_c = 1$. For $T = 0$ we get the equation $m = \text{erf}[m/\sqrt{2\alpha}]$, giving a continuous transition to $m > 0$ solutions at $\alpha_c = 2/\pi \approx 0.637$. The remaining question concerns the nature of the $m = 0$ state. Inserting $m(t) = \theta(t) = 0$ (for all t) into $C(t, t')$ tells us that $C(t, t') = f[C(t-1, t'-1)]$ for $t > t' > 0$, with ‘initial conditions’ $C(t, 0) = m(t)m_0$, where

$$f[C] = \int \frac{d\phi_a d\phi_b}{2\pi\sqrt{1-C^2}} e^{-\frac{1}{2} \frac{\phi_a^2 + \phi_b^2 - 2C\phi_a\phi_b}{1-C^2}} \tanh[\beta\sqrt{\alpha}\phi_a] \tanh[\beta\sqrt{\alpha}\phi_b].$$

In the $m = 0$ regime we have $C(t, 0) = 0$ for any $t > 0$, inducing $C(t, t') = 0$ for any $t > t'$, due to $f[0] = 0$. Thus we conclude that $C(t, t') = \delta_{t, t'}$ in the $m = 0$ phase, i.e., this phase is para-magnetic rather than of a spin-glass type.

On the other hand, physics of networks with symmetric dilution is more complicated situation. In spite of the extreme dilution, the interaction symmetry makes sure that the spins still have a sufficient number of common ancestors for complicated correlations to build up in finite time. We have

$$\begin{aligned}
h(t|\{\sigma\}, \{\phi\}) &= m(t) + \theta(t) + \alpha \sum_{t' < t} G(t, t') \sigma(t') + \alpha^{\frac{1}{2}} \phi(t), \\
P[\{\phi\}] &= \frac{e^{-\frac{1}{2} \sum_{t, t'} \phi(t) \mathbf{C}^{-1}(t, t') \phi(t')}}{(2\pi)^{(t_m+1)/2} \det^{\frac{1}{2}} \mathbf{C}}.
\end{aligned}$$

Thus the effective single neuron problem is found to be exactly of the form found also for the Gaussian model defined above (which, in turn, maps onto

the parallel dynamics *Sherrington–Kirkpatrick model* (SK model) [SK75]) with the synapses $J_{ij} = J_0 \xi_i \xi_j / N + J z_{ij} / \sqrt{N}$ (in which the z_{ij} are symmetric zero-average and unit-variance Gaussian variables, and $J_{ii} = 0$ for all i), with the identification:

$$J \rightarrow \sqrt{\alpha}, \quad \text{with} \quad J_0 \rightarrow 1.$$

Since one can show that for $J_0 > 0$ the parallel dynamics SK model gives the same equilibrium state as the sequential one, we can now immediately write down the stationary solution of our dynamic equations which corresponds to the FDT regime, with $q = \lim_{\tau \rightarrow \infty} \lim_{t \rightarrow \infty} C(t, t + \tau)$:

$$q = \int Dz \tanh^2[\beta(m + z\sqrt{\alpha q})], \quad m = \int Dz \tanh[\beta(m + z\sqrt{\alpha q})].$$

3.5.3 Hierarchical Self-Programming in Neural Nets

The *hierarchical self-programming* in recurrent neural networks was studied in [UC02]. The authors studied recurrent networks of binary (Ising-spin) neuronal state variables with symmetric interactions J_{ij} , taken to be of infinite range. In contrast to most standard neural network models, not only the neuron states but also the interactions were allowed to evolve in time (simultaneously), driven by correlations in the states of the neurons (albeit slowly compared to the dynamics of the latter), reflecting the effect of ‘learning’ or ‘long-term potentiation’ in real nervous tissue. Since the interactions represent the ‘programme’ of the system, and since the slow interaction dynamics are driven by the states of the neurons (the ‘processors’), such models can be regarded as describing self-programming information-processing systems, which can be expected to exhibit highly complex dynamical behavior.

The first papers in which self-programming recurrent neural networks were studied appear to be [Shi87, DH85]. In the language of self-programming systems one could say that these authors were mostly concerned with the stability properties of embedded ‘programmes’ (usually taken to be those implementing content-addressable or associative memories). In both [Shi87, DH85] the programme dynamics, i.e., that of the $\{J_{ij}\}$, was defined to be adiabatically slow compared to the neuronal dynamics, and fully deterministic. However, the authors already made the important observation that the natural type of (deterministic) programme dynamics (from a biological point of view), so-called Hebbian learning, could be written as a gradient descent of the interactions $\{J_{ij}\}$ on the free energy surface of a symmetric recurrent neural network equipped with these interactions.

In order to study more generally the potential of such self-programming systems, several authors (simultaneously and independently) took the natural next step [CPS93, PCS93, DFM94, PS94, FD94]: they generalized the interaction dynamics by adding Gaussian white noise to the deterministic laws, converting the process into one described by conservative Langevin equations,

and were thus able to set up an equilibrium statistical mechanics of the self-programming process. This was (surprisingly) found to take the form of a replica theory with finite replica dimension, whose value was given by the ratio of the noise levels in the neuronal dynamics and the interaction dynamics, respectively. Furthermore, adding explicit quenched disorder to the problem in the form of additional random (but frozen) forces in the interaction dynamics, led to theories with two nested levels of replicas, one representing the disorder (with zero replica dimension) and one representing the adiabatically slow dynamics of the interactions [PS94, JBC98] (with nonzero replica dimension). The structure of these latter theories was found to be more or less identical to those of ordinary disordered spin systems such as the SK model [SK75], with fixed interactions but quenched disorder, when described by replica theories with one step replica symmetry breaking (RSB) [MPV87]. The only (yet crucial) difference was that in ordinary disordered spin systems the size m of the level-1 block in the Parisi solution is determined by extermination of the free energy, which forces m to lie in the interval $[0, 1]$, whereas in the self-programming neural networks of [PS94, JBC98] m was an independent control parameters, given by the ratio of two temperatures, which can take any non-zero value. As a consequence one can observe in the latter systems, for sufficiently large values of such dimensions, much more complicated scenarios of (generally discontinuous) phase transitions.

In contrast to the previous models involving coupled dynamics of fast neurons and slow interactions, [UC02] studied systems in which the interactions did not evolve on a single time-scale, but where they were divided into a hierarchy of L different groups, each with their own characteristic time-scale τ_ℓ and noise level T_ℓ ($\ell = 1, \dots, L$), describing a hierarchy of increasingly non-volatile programming levels. This appeared to be a much more realistic representation of self-programming systems; conventional programmes generally take the form of hierarchies of routines, sub-routines and so on, and it would appear appropriate to allow low-level sub-routines to be more easily modifiable than high-level ones. In order to retain analytical solvability we choose the different groups of interactions randomly (prescribing only their sizes).

The authors of [UC02] solved the model in equilibrium, and found, upon making the replica-symmetric (i.e., ergodic) ansatz within each level of our hierarchy, a theory which resembled, but was not identical to Parisi's L -level replica symmetry-breaking solution for spin systems with frozen disorder. They referred to the binary neurons as spins and to the synaptic interactions as couplings. They wrote the N -spin state vector as $\boldsymbol{\sigma} = (\sigma_1, \dots, \sigma_N) \in \{-1, 1\}^N$, and the matrix of interactions as $\mathbf{J} = \{J_{ij}\}$. The spins were taken to have a stochastic Glauber-type dynamics such that for *stationary* choices of the couplings the microscopic spin probability density would evolve towards a Boltzmann distribution

$$p_\infty(\boldsymbol{\sigma}) = \frac{e^{-\beta H(\boldsymbol{\sigma})}}{Z}, \quad Z = \sum_{\boldsymbol{\sigma}} e^{-\beta H(\boldsymbol{\sigma})}, \quad (3.434)$$

with the conventional Hamiltonian

$$H(\boldsymbol{\sigma}) = -J_{ij}\sigma_i\sigma_j, \quad (3.435)$$

where $i, j \in \{1, \dots, N\}$, and with the inverse temperature $\beta = T^{-1}$.

The couplings J_{ij} also evolve in a stochastic manner, in response to the states of the spins, but adiabatically slowly compared to the spins, such that on the time-scales of the couplings the spins are always in an equilibrium state described by (3.434). For the coupling dynamics the following Langevin equations are proposed:

$$\tau_{ij}\dot{J}_{ij} = \frac{1}{N}\langle\sigma_i\sigma_j\rangle_{\text{sp}} - \mu_{ij}J_{ij} + \eta_{ij}(t), \quad (i < j = 1 \dots N), \quad (3.436)$$

with $\tau_{ij} \gg 1$. In the adiabatic limit $\tau_{ij} \rightarrow \infty$ the term $\langle\sigma_i\sigma_j\rangle_{\text{sp}}$, representing spin correlations associated with the coupling J_{ij} , becomes an average over the Boltzmann distribution (3.434) of the spins, given the instantaneous couplings $\mathbf{J}$. The $\eta_{ij}(t)$ represent Gaussian white noise contributions, of zero mean and covariance

$$\langle\eta_{ij}(t)\eta_{kl}(t')\rangle = 2T_{ij}\delta_{ik}\delta_{jl}\delta(t-t'),$$

with associated temperature $T_{ij} = \beta_{ij}^{-1}$. Appropriate factors of N have been introduced in order to ensure non-trivial behavior in the limit $N \rightarrow \infty$. We classify the spin pairs (i, j) according to the characteristic time-scale τ_{ij} and the control parameters (T_{ij}, μ_{ij}) associated with their interactions J_{ij} . In contrast to papers such as [CPS93, PCS93, PS94, JBC98], where $\tau_{ij} = \tau$, $T_{ij} = \tilde{T}$ and $\mu_{ij} = \mu$ for all (i, j) , here the various time-scales, temperatures and decay rates are no longer assumed to be identical, but to come in L distinct adiabatically separated groups I_ℓ (always with $i < j$):

$$I_\ell = \{(i, j) \mid \tau_{ij} = \tau_\ell, \quad T_{ij} = T_\ell, \mu_{ij} = \mu_\ell\}, \quad (l = 1, \dots, L),$$

with $1 \ll \tau_1 \ll \tau_2 \ll \dots \ll \tau_L$. Thus $\{(i, j)\} = \bigcup_{\ell \leq L} I_\ell$. We will write the set of spin-interactions with time-scale τ_ℓ as $\mathbf{J}^\ell = \{J_{ij} \mid (i, j) \in I_\ell\}$. The interactions in group I_2 are adiabatically slow compared to those in group I_1 , and so on. The rationale of this set-up is that, in information processing terms, this would represent a stochastic self-programming neural information processing system equipped with a program which consists of a hierarchy of increasingly less volatile and less easily modifiable sub-routines.

Finally we have to define the detailed partitioning of the $\frac{1}{2}N(N-1)$ interactions into the L volatility groups. We introduce $\epsilon_{ij}(\ell) \in \{0, 1\}$ such that $\epsilon_{ij}(\ell) = 1$ iff $(i, j) \in I_\ell$, so $\sum_{\ell=1}^L \epsilon_{ij}(\ell) = 1$ for all (i, j) . In order to arrive at a solvable mean-field problem, with full equivalence of the N sites, we will

choose the $\epsilon_{ij}(\ell)$ independently at random for each pair (i, j) with $i < j$, with probabilities [UC02]

$$\text{Prob}[\epsilon_{ij}(\ell) = 1] = \epsilon_\ell, \quad \sum_{\ell=1}^L \epsilon_\ell = 1. \quad (3.437)$$

These time-scale and temperature allocation variables $\{\epsilon_{ij}(\ell)\}$ thus introduce quenched disorder into the problem in hand.

Let us denote averages over the probability distribution of the couplings at level ℓ in the hierarchy (for which $\tau_{ij} = \tau_\ell$, $T_{ij} = T_\ell$ and $\mu_{ij} = \mu_\ell$) as $\langle \dots \rangle_\ell$. At every level ℓ , the stochastic equation (3.436) for those couplings which evolve on that particular time-scale τ_ℓ has now become conservative:

$$\begin{aligned} \tau_\ell \dot{J}_{ij} &= \frac{1}{N} \langle \dots \langle \langle \sigma_i \sigma_j \rangle_{\text{sp}} \rangle_{\ell-1} \dots \rangle_{\ell-1} - \mu_\ell J_{ij} + \eta_{ij}(t) \sqrt{\frac{\tau_\ell}{N}} \\ &= -\frac{1}{N} \frac{\partial}{\partial J_{ij}} H_\ell(\mathbf{J}^\ell, \dots, \mathbf{J}^L) + \eta_{ij}(t) \sqrt{\frac{\tau_\ell}{N}}, \end{aligned} \quad (3.438)$$

with the following effective Hamiltonian for the couplings at level ℓ :

$$H_1(\mathbf{J}^1, \dots, \mathbf{J}^L) = -\beta^{-1} \log Z[\mathbf{J}^1, \dots, \mathbf{J}^L], \quad (3.439)$$

$$H_{\ell+1}(\mathbf{J}^{\ell+1}, \dots, \mathbf{J}^L) = -\beta_\ell^{-1} \log Z_\ell[\mathbf{J}^{\ell+1}, \dots, \mathbf{J}^L] \quad (1 \leq \ell < L), \quad (3.440)$$

and with the partition functions

$$Z[\mathbf{J}^1, \dots, \mathbf{J}^L] = \sum_{\boldsymbol{\sigma}} e^{-\beta H(\boldsymbol{\sigma}, \mathbf{J})}, \quad (3.441)$$

$$Z_\ell[\mathbf{J}^{\ell+1}, \dots, \mathbf{J}^L] = \int d\mathbf{J}^\ell e^{-\beta_\ell H_\ell(\mathbf{J}^\ell, \dots, \mathbf{J}^L)}, \quad (1 \leq \ell < L), \quad (3.442)$$

$$Z_L = \int d\mathbf{J}^L e^{-\beta_L H_L(\mathbf{J}^L)}, \quad \text{in which} \quad (3.443)$$

$$H(\boldsymbol{\sigma}, \mathbf{J}) = -J_{ij} \sigma_i \sigma_j + \frac{1}{2} N \mu_{ij} J_{ij}^2 \quad (3.444)$$

This describes a hierarchy of nested equilibrations. At each time-scale τ_ℓ the interactions $\mathbf{J}^\ell$ equilibrate to a Boltzmann distribution, with an effective Hamiltonian H_ℓ which is the free energy of the previous (faster) level $\ell + 1$ in the hierarchy, starting from the overall Hamiltonian (3.444) (for spins and couplings) at the fastest (spin) level. As a result of having different effective temperatures T_ℓ associated with each level, the partition functions of subsequent levels are found to generate replica theories with replica dimensions $m_\ell \geq 0$ which represent the ratios of the effective temperatures of the two levels involved. This follows from substitution of (3.440) into (3.442):

$$Z_\ell[\mathbf{J}^{\ell+1}, \dots, \mathbf{J}^L] = \int d\mathbf{J}^\ell \{Z_{\ell-1}[\mathbf{J}^\ell, \dots, \mathbf{J}^L]\}^{m_\ell}, \quad (3.445)$$

$$m_\ell = \beta_\ell / \beta_{\ell-1} \quad m_1 = \beta_1 / \beta \quad (3.446)$$

The statics of the system, including the effect of the quenched disorder, are governed by the disorder-averaged free energy $\mathcal{F}$ associated with the partition function Z_L in (3.443), where the slowest variables have finally been integrated out [UC02]:

$$\mathcal{F} = -\frac{1}{\beta_L} \overline{\log Z_L} = -\lim_{m_{L+1} \rightarrow 0} \frac{1}{m_{L+1} \beta_L} \log \overline{Z_L^{m_{L+1}}}. \quad (3.447)$$

This function is found to act as the general generator of equilibrium values for observables at any level in the hierarchy, since upon adding suitable partition terms to the Hamiltonian (3.444), i.e., $H(\boldsymbol{\sigma}, \mathbf{J}) \rightarrow H(\boldsymbol{\sigma}, \mathbf{J}) + \lambda \Phi(\boldsymbol{\sigma}, \mathbf{J})$, one finds:

$$\overline{\langle \dots \langle \Phi(\boldsymbol{\sigma}, \mathbf{J}) \rangle_{\text{sp}} \dots \rangle_L} = \lim_{\lambda \rightarrow 0} \frac{\partial}{\partial \lambda} \mathcal{F}.$$

We can now combine our previous results and write down an explicit expression for $\mathcal{F}$, involving multiple replications due to (3.445, 3.447). We find $\mathcal{F} = \lim_{m_{L+1} \rightarrow 0} F[m_1, \dots, m_{L+1}]$, with $F[m_1, \dots, m_{L+1}] = -(m_{L+1} \beta_L)^{-1} \log \overline{Z_1^{m_{L+1}}}$, and where $Z_L^{m_{L+1}}$ is written as [UC02]

$$\begin{aligned} Z_L^{m_{L+1}} &= \int \left[\prod_{\alpha_{L+1}} d\mathbf{J}^{L, \alpha_{L+1}} \right] \prod_{\alpha_{L+1}} \{Z_{L-1}[\mathbf{J}^{L, \alpha_{L+1}}]\}^{m_L} \\ &= \int \left[\prod_{\alpha_{L+1}} d\mathbf{J}^{L, \alpha_{L+1}} \dots \prod_{\alpha_2, \dots, \alpha_{L+1}} d\mathbf{J}^{1, \alpha_2, \dots, \alpha_{L+1}} \right] \\ &\quad \times \prod_{\alpha_2, \dots, \alpha_{L+1}} \{Z[\mathbf{J}^{1, \alpha_2, \dots, \alpha_{L+1}}, \dots, \mathbf{J}^{L, \alpha_{L+1}}]\}^{m_1} \\ &= \sum_{\{\boldsymbol{\sigma}^{\alpha_1, \dots, \alpha_{L+1}}\}} \int \left[\prod_{\alpha_{L+1}} d\mathbf{J}^{L, \alpha_{L+1}} \dots \prod_{\alpha_2, \dots, \alpha_{L+1}} d\mathbf{J}^{1, \alpha_2, \dots, \alpha_{L+1}} \right] \\ &\quad \times \prod_{\alpha_1, \dots, \alpha_{L+1}} e^{-\beta H(\boldsymbol{\sigma}^{\alpha_1, \dots, \alpha_{L+1}}, \boldsymbol{\sigma}^{1, \alpha_2, \dots, \alpha_{L+1}}, \dots, \boldsymbol{\sigma}^{L, \alpha_{L+1}})} \\ &= \sum_{\{\boldsymbol{\sigma}^{\alpha_1, \dots, \alpha_{L+1}}\}} \prod_{\ell=1}^L \prod_{(i < j) \in I_\ell} \prod_{\alpha_{\ell+1}, \dots, \alpha_{L+1}} \\ &\quad \times \int dz e^{-\frac{1}{2} \beta \mu_\ell N [m_1 \dots m_\ell z^2 - 2(\mu_\ell N)^{-1} z \sum_{\alpha_1 \dots \alpha_\ell} \sigma_i^{\alpha_1, \dots, \alpha_{L+1}} \sigma_j^{\alpha_1, \dots, \alpha_{L+1}}]} \\ &= \sum_{\{\boldsymbol{\sigma}^{\alpha_1, \dots, \alpha_{L+1}}\}} e^{\frac{\beta}{2N} \sum_{\ell \leq L} \frac{1}{m_1 \dots m_\ell \mu_\ell} \sum_{(i < j) \in I_\ell} \sum_{\alpha_{\ell+1}, \dots, \alpha_{L+1}} \\ &\quad [\sum_{\alpha_1 \dots \alpha_\ell} \sigma_i^{\alpha_1, \dots, \alpha_{L+1}} \sigma_j^{\alpha_1, \dots, \alpha_{L+1}}]}. \end{aligned}$$

(modulo irrelevant constants), in which always $\alpha_\ell = 1, \dots, m_\ell$.

Now, in order to average the last expression for $Z_L^{m_{L+1}}$ over the disorder, we note that the spin summation in the exponent can also be written in terms of the allocation variables $\epsilon_{ij}(\ell)$:

$$\begin{aligned} \sum_{\ell \leq L} \frac{1}{m_1 \cdots m_\ell \mu_\ell} \sum_{(i < j) \in I_\ell} \sum_{\alpha_{\ell+1}, \dots, \alpha_{L+1}} \dots = \\ \sum_{i < j} \sum_{\ell \leq L} \frac{\beta}{\beta_\ell} \frac{\epsilon_{ij}(\ell)}{\mu_\ell} \sum_{\alpha_{\ell+1}, \dots, \alpha_{L+1}} \left[\sum_{\alpha_1 \dots \alpha_\ell} \sigma_i^{\alpha_1, \dots, \alpha_{L+1}} \sigma_j^{\alpha_1, \dots, \alpha_{L+1}} \right]^2, \end{aligned}$$

where we used $m_1 \dots m_\ell = \beta_\ell / \beta$. This allows us to carry out the average over the (independent) $\epsilon_{ij}(\ell)$ in:

$$\begin{aligned} F[\dots] &= -\frac{1}{m_{L+1} \beta_L} \log \sum_{\{\sigma^{\alpha_1, \dots, \alpha_{L+1}}\}} \\ &\quad \times \prod_{i < j} \left[e^{\frac{\beta^2}{2N} \sum_{\ell \leq L} \frac{\epsilon_{ij}(\ell)}{\beta_\ell \mu_\ell} \sum_{\alpha_{\ell+1}, \dots, \alpha_{L+1}} \left[\sum_{\alpha_1 \dots \alpha_\ell} \sigma_i^{\alpha_1, \dots, \alpha_{L+1}} \sigma_j^{\alpha_1, \dots, \alpha_{L+1}} \right]^2} \right] \\ &= -\frac{1}{m_{L+1} \beta_L} \log \sum_{\{\sigma^{\alpha_1, \dots, \alpha_{L+1}}\}} e^{\mathcal{O}(N^0)} \\ &\quad \times \prod_{i < j} \left[1 + \frac{\beta^2}{2N} \sum_{\ell \leq L} \frac{\overline{\epsilon_{ij}(\ell)}}{\beta_\ell \mu_\ell} \sum_{\alpha_{\ell+1}, \dots, \alpha_{L+1}} \left[\sum_{\alpha_1 \dots \alpha_\ell} \sigma_i^{\alpha_1, \dots, \alpha_{L+1}} \sigma_j^{\alpha_1, \dots, \alpha_{L+1}} \right]^2 \right] \\ &= -\frac{1}{m_{L+1} \beta_L} \log \sum_{\{\sigma^{\alpha_1, \dots, \alpha_{L+1}}\}} e^{\mathcal{O}(N^0)} \\ &\quad \times \exp \left\{ \frac{N \beta^2}{4} \sum_{\alpha_1, \dots, \alpha_{L+1}} \sum_{\beta_1, \dots, \beta_{L+1}} \sum_{\ell \leq L} \frac{\epsilon_\ell}{\beta_\ell \mu_\ell} \delta_{(\alpha_{\ell+1}, \dots, \alpha_{L+1}), (\beta_{\ell+1}, \dots, \beta_{L+1})} \right. \\ &\quad \left. \times \left[\frac{1}{N} \sum_i \sigma_i^{\alpha_1, \dots, \alpha_{L+1}} \sigma_i^{\beta_1, \dots, \beta_{L+1}} \right]^2 \right\}, \end{aligned}$$

(again modulo irrelevant constants). We abbreviate $a = (\alpha_1, \dots, \alpha_{L+1})$, and introduce the spin-glass replica order parameters $q_{ab} = \frac{1}{N} \sigma_i^{\alpha_1, \dots, \alpha_{L+1}} \sigma_i^{\beta_1, \dots, \beta_{L+1}}$ by inserting appropriate integrals over δ -distributions, and arrive at

$$\begin{aligned} F[\dots] &= -\frac{1}{m_{L+1} \beta_L} \log \int \{dq_{ab} d\hat{q}_{ab}\} e^{NG[\{q_{ab}, \hat{q}_{ab}\}] + \mathcal{O}(N^0)} \\ G[\{q_{ab}, \hat{q}_{ab}\}] &= i \sum_{ab} \hat{q}_{ab} q_{ab} + \frac{\beta^2}{4} \sum_{\ell \leq L} \frac{\epsilon_\ell}{\beta_\ell \mu_\ell} \sum_{ab} \delta_{(\alpha_{\ell+1}, \dots, \alpha_{L+1}), (\beta_{\ell+1}, \dots, \beta_{L+1})} q_{ab}^2 \\ &\quad + \log \sum_{\{\sigma_{\alpha_1, \dots, \alpha_{L+1}}\}} e^{-i \sum_{ab} \hat{q}_{ab} \sigma_{\alpha_1, \dots, \alpha_{L+1}} \sigma_{\beta_1, \dots, \beta_{L+1}}}. \end{aligned}$$

For $N \rightarrow \infty$ the above integral can be evaluated by steepest descent, and upon elimination of the conjugate order parameters $\{\hat{q}_{ab}\}$ by variation of $\{q_{ab}\}$, the disorder-averaged free energy per spin $f = \lim_{N \rightarrow \infty} \mathcal{F}/N$ is found to be

$$f = \lim_{m_{L+1} \rightarrow 0} \frac{1}{m_{L+1}\beta_L} \left\{ \frac{\beta^2}{4} \sum_{\ell \leq L} \frac{\epsilon_\ell}{\beta_\ell \mu_\ell} \sum_{ab} \delta_{(\alpha_{\ell+1}, \dots, \alpha_{L+1}), (\beta_{\ell+1}, \dots, \beta_{L+1})} q_{ab}^2 \right. \\ \left. - \log \sum_{\{\sigma_a\}} e^{\frac{\beta^2}{2} \sum_{\ell \leq L} \frac{\epsilon_\ell}{\beta_\ell \mu_\ell} \sum_{ab} \delta_{(\alpha_{\ell+1}, \dots, \alpha_{L+1}), (\beta_{\ell+1}, \dots, \beta_{L+1})} q_{ab} \sigma_a \sigma_b} \right\}. \quad (3.448)$$

The saddle-point equations from which to solve the $\{q_{ab}\}$ are given by [UC02]

$$q_{cd} = \frac{\sum_{\{\sigma_a\}} \sigma_c \sigma_d e^{\frac{\beta^2}{2} \sum_{\ell \leq L} \frac{\epsilon_\ell}{\beta_\ell \mu_\ell} \sum_{ab} \delta_{(\alpha_{\ell+1}, \dots, \alpha_{L+1}), (\beta_{\ell+1}, \dots, \beta_{L+1})} q_{ab} \sigma_a \sigma_b}}{\sum_{\{\sigma_a\}} e^{\frac{\beta^2}{2} \sum_{\ell \leq L} \frac{\epsilon_\ell}{\beta_\ell \mu_\ell} \sum_{ab} \delta_{(\alpha_{\ell+1}, \dots, \alpha_{L+1}), (\beta_{\ell+1}, \dots, \beta_{L+1})} q_{ab} \sigma_a \sigma_b}}.$$

with our short-hand $a = (\alpha_1, \dots, \alpha_{L+1})$, and with $\alpha_\ell \in \{1, \dots, m_\ell\}$ for all ℓ , where the dimensions m_ℓ are given by the ratios of the temperatures of subsequent programming levels in the hierarchy according to (3.446).

Our full order parameters are $q_{ab} = \frac{1}{N} \sigma_i^{\alpha_1, \dots, \alpha_{L+1}} \sigma_i^{\beta_1, \dots, \beta_{L+1}}$. Note that $\alpha_{L+1} = \beta_{L+1}$. Spin variables with $\alpha_L = \beta_L$ have identical level- L bonds. Those with $\alpha_L = \beta_L$ and $\alpha_{L-1} = \beta_{L-1}$ have identical level L bonds *and* identical level $L-1$ bonds, and so on. Hence, for the present model the replica-symmetric (RS) ansatz (describing ergodicity at each level of the hierarchy of time-scales) takes the following form [UC02]:

$$\begin{aligned} \alpha_L \neq \beta_L : \quad q_{ab} &= q_L, \\ (\alpha_{\ell+1}, \dots, \alpha_{L+1}) = (\beta_{\ell+1}, \dots, \beta_{L+1}), \quad \alpha_\ell \neq \beta_\ell : \quad q_{ab} &= q_\ell, \\ a = b : \quad q_{ab} &= 1, \end{aligned}$$

$$\text{or} \quad q_{ab} = \sum_{\ell=1}^L q_\ell \delta_{(\alpha_{\ell+1}, \dots, \alpha_{L+1}), (\beta_{\ell+1}, \dots, \beta_{L+1})} \bar{\delta}_{\alpha_\ell \beta_\ell} + \delta_{ab}, \quad (3.449)$$

where $\bar{\delta}_{ij} = 1 - \delta_{ij}$, and where $0 \leq q_L \leq \dots \leq q_1 \leq 1$.

3.5.4 Neural Path-Integral Model for the Cerebellum

Recall that human motion is naturally driven by synergistic action of more than 600 skeletal muscles. While the muscles generate driving torques in the moving joints, subcortical neural system performs both local and global (loco)motion control: first reflexly controlling contractions of individual muscles, and then orchestrating all the muscles into synergetic actions in order to produce efficient movements. While the local reflex control of individual

muscles is performed on the *spinal control level*, the global integration of all the muscles into coordinated movements is performed within the *cerebellum*.

All hierarchical subcortical neuro-muscular physiology, from the bottom level of a single muscle fiber, to the top level of cerebellar muscular synergy, acts as a *temporal* $\langle out|in \rangle$ *reaction*, in such a way that the higher level acts as a command/control space for the lower level, itself representing an abstract image of the lower one:

1. At the *muscular level*, we have *excitation-contraction dynamics* [Hat77a, Hat78, Hat77b], in which $\langle out|in \rangle$ is given by the following sequence of nonlinear diffusion processes: *neural-action-potential* $\rightsquigarrow$ *synaptic-potential* $\rightsquigarrow$ *muscular-action-potential* $\rightsquigarrow$ *excitation-contraction-coupling* $\rightsquigarrow$ *muscle-tension-generating* [II06a, II05]. Its purpose is the generation of muscular forces, to be transferred into driving torques within the joint anatomical geometry.
2. At the *spinal level*, $\langle out|in \rangle$ is given by *autogenetic-reflex stimulus-response control* [Hou79]. Here we have a neural image of all individual muscles. The main purpose of the spinal control level is to give both positive and negative feedbacks to stabilize generated muscular forces within the ‘homeostatic’ (or, more appropriately, ‘homeokinetic’) limits. The individual muscular actions are combined into flexor-extensor (or agonist-antagonist) pairs, mutually controlling each other. This is the mechanism of *reciprocal innervation of agonists and inhibition of antagonists*. It has a purely mechanical purpose to form the so-called *equivalent muscular actuators* (EMAs), which would generate driving torques $T_i(t)$ for all movable joints.
3. At the *cerebellar level*, $\langle out|in \rangle$ is given by *sensory-motor integration* [HBB96]. Here we have an abstracted image of all autogenetic reflexes. The main purpose of the cerebellar control level is integration and fine tuning of the action of all active EMAs into a synchronized movement, by *supervising* the individual autogenetic reflex circuits. At the same time, to be able to perform in new and unknown conditions, the cerebellum is continuously adapting its own neural circuitry by unsupervised (self-organizing) learning. Its action is subconscious and automatic, both in humans and in animals.

Naturally, we can ask the question: Can we assign a single $\langle out|in \rangle$ measure to all these neuro-muscular stimulus-response reactions? We think that we can do it; so in this Letter, we present the concept of *adaptive sensory-motor transition amplitude* as a unique measure for this temporal $\langle out|in \rangle$ relation. Conceptually, this $\langle out|in \rangle$ -*amplitude* can be formulated as the ‘*neural path integral*’:

$$\langle out|in \rangle \equiv \langle motor|sensory \rangle_{\text{amplitude}} = \int \mathcal{D}[w, x] e^{iS[x]}. \quad (3.450)$$

Here, the integral is taken over all *activated* (or, ‘fired’) *neural pathways* $x^i = x^i(t)$ of the cerebellum, connecting its input *sensory*–state with its output *motor*–state, symbolically described by *adaptive neural measure* $\mathcal{D}[w, x]$, defined by the weighted product (of discrete time steps)

$$\mathcal{D}[w, x] = \lim_{n \rightarrow \infty} \prod_{t=1}^n w^i(t) dx^i(t),$$

in which the *synaptic weights* $w^i = w^i(t)$, included in all active neural pathways $x^i = x^i(t)$, are updated by the unsupervised *Hebbian-like learning* rule

$$w^i(t+1) = w^i(t) + \frac{\sigma}{\eta} (w_d^i(t) - w_a^i(t)), \quad (3.451)$$

where $\sigma = \sigma(t)$, $\eta = \eta(t)$ represent local neural *signal* and *noise* amplitudes, respectively, while superscripts d and a denote *desired* and *achieved* neural states, respectively. Theoretically, equations (3.450–3.451) define an ∞ –*dimensional neural network*. Practically, in a computer simulation we can use $10^7 \leq n \leq 10^8$, roughly corresponding to the number of neurons in the cerebellum.

The exponent term $S[x]$ in equation (3.450) represents the *autogenetic-reflex action*, describing reflexly-induced motion of all active EMAs, from their initial *stimulus*–state to their final *response*–state, along the family of extremal (i.e., Euler–Lagrangian) paths $x_{\min}^i(t)$. ($S[x]$ is properly derived in (3.454–3.455) below.)

Spinal Autogenetic Reflex Control

Recall (from Introduction) that at the spinal control level we have the autogenetic reflex *motor servo* [Hou79], providing the local, reflex feedback loops for individual muscular contractions. A voluntary contraction force F of human skeletal muscle is reflexly excited (positive feedback $+F^{-1}$) by the responses of its *spindle receptors* to stretch and is reflexly inhibited (negative feedback $-F^{-1}$) by the responses of its *Golgi tendon organs* to contraction. Stretch and unloading reflexes are mediated by combined actions of several autogenetic neural pathways, forming the *motor servo*.

In other words, branches of the afferent fibers also synapse with interneurons that inhibit motor neurons controlling the antagonistic muscles – *reciprocal inhibition*. Consequently, the stretch stimulus causes the antagonists to relax so that they cannot resist the shortening of the stretched muscle caused by the main reflex arc. Similarly, firing of the Golgi tendon receptors causes inhibition of the muscle contracting too strong and simultaneous *reciprocal activation* of its antagonist. Both mechanisms of reciprocal inhibition and activation performed by the autogenetic circuits $+F^{-1}$ and $-F^{-1}$, serve to generate the well-tuned EMA-driving torques T_i .

Now, once we have properly defined the symplectic musculo-skeletal dynamics [Iva04] on the biomechanical (momentum) phase-space manifold T^*M^N , we can proceed in formalizing its hierarchical subcortical neural control. By introducing the *coupling Hamiltonians* $H^m = H^m(q, p)$, selectively corresponding *only* to the $M \leq N$ *active joints*, we define the *affine Hamiltonian control function* $H_{aff} : T^*M^N \rightarrow \mathbb{R}$, in local canonical coordinates on T^*M^N given by (adapted from [NS90] for the biomechanical purpose)

$$H_{aff}(q, p) = H_0(q, p) - H^m(q, p) T_m, \quad (m = 1, \dots, M \leq N), \quad (3.452)$$

where $T_m = T_m(t, q, p)$ are affine feedback torque one-forms, different from the initial driving torques T_i acting in all the joints. Using the affine Hamiltonian function (3.452), we get the *affine Hamiltonian servo-system* [Iva04],

$$\begin{aligned} \dot{q}^i &= \frac{\partial H_0(q, p)}{\partial p_i} - \frac{\partial H^m(q, p)}{\partial p_i} T_m, \\ \dot{p}_i &= -\frac{\partial H_0(q, p)}{\partial q^i} + \frac{\partial H^m(q, p)}{\partial q^i} T_m, \\ q^i(0) &= q_0^i, \quad p_i(0) = p_i^0, \quad (i = 1, \dots, N; \quad m = 1, \dots, M \leq N). \end{aligned} \quad (3.453)$$

The affine Hamiltonian control system (3.453) gives our formal description for the autogenetic spinal motor-servo for all $M \leq N$ activated (i.e., working) EMAs.

Cerebellum – the Comparator

Having, thus, defined the spinal reflex control level, we proceed to model the top subcortical commander/controller, the *cerebellum*. It is a brain region anatomically located at the bottom rear of the head (the hindbrain), directly above the brainstem, which is important for a number of subconscious and automatic motor functions, including motor learning. It processes information received from the motor cortex, as well as from proprioceptors and visual and equilibrium pathways, and gives ‘instructions’ to the motor cortex and other subcortical motor centers (like the basal nuclei), which result in proper balance and posture, as well as smooth, coordinated skeletal movements, like walking, running, jumping, driving, typing, playing the piano, etc. Patients with cerebellar dysfunction have problems with precise movements, such as walking and balance, and hand and arm movements. The cerebellum looks *similar in all animals*, from fish to mice to humans. This has been taken as evidence that it performs a common function, such as regulating motor learning and the timing of movements, in all animals. Studies of simple forms of motor learning in the vestibulo-ocular reflex and eye-blink conditioning are demonstrating that timing and amplitude of learned movements are encoded by the cerebellum.

The cerebellum is responsible for coordinating precisely timed $\langle out|in \rangle$ activity by integrating motor output with ongoing sensory feedback. It receives extensive projections from sensory-motor areas of the cortex and the periphery and directs it back to premotor and motor cortex [Ghe90, Ghe91]. This suggests a role in sensory-motor integration and the timing and execution of human movements. The cerebellum stores patterns of motor control for frequently performed movements, and therefore, its circuits are changed by experience and training. It was termed the *adjustable pattern generator* in the work of J. Houk and collaborators [HBB96]. Also, it has become the inspiring ‘brain-model’ in the recent robotic research [SA98, Sch98].

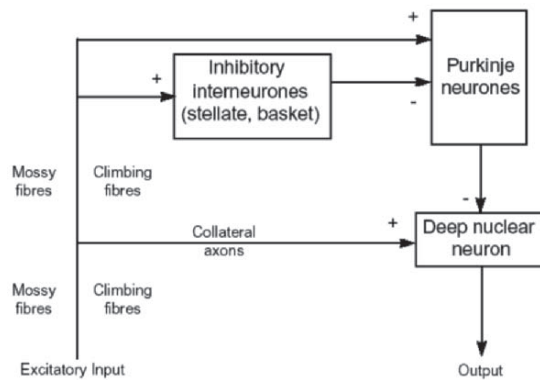


Fig. 3.5. Schematic $\langle out|in \rangle$ organization of the primary cerebellar circuit. In essence, excitatory inputs, conveyed by collateral axons of Mossy and Climbing fibers activate directly neurones in the Deep cerebellar nuclei. The activity of these latter is also modulated by the inhibitory action of the cerebellar cortex, mediated by the Purkinje cells.

Comparing the number of its neurons ($10^7 - 10^8$), to the size of conventional neural networks, suggests that artificial neural nets *cannot* satisfactorily model the function of this sophisticated ‘super-bio-computer’, as its dimensionality is virtually infinite. Despite a lot of research dedicated to its structure and function (see [HBB96] and references there cited), the real nature of the cerebellum still remains a ‘mystery’.

The main function of the cerebellum as a motor controller is depicted in Figure 3.6. A coordinated movement is easy to recognize, but we know little about how it is achieved. In search of the neural basis of coordination, a model of spinocerebellar interactions was recently presented in [AG05], in which the structural and functional organizing principle is a division of the cerebellum into discrete micro-complexes. Each micro-complex is the recipient of a specific motor error signal - that is, a signal that conveys information about an inappropriate movement. These signals are encoded by spinal reflex circuits and conveyed to the cerebellar cortex through climbing fibre afferents. This

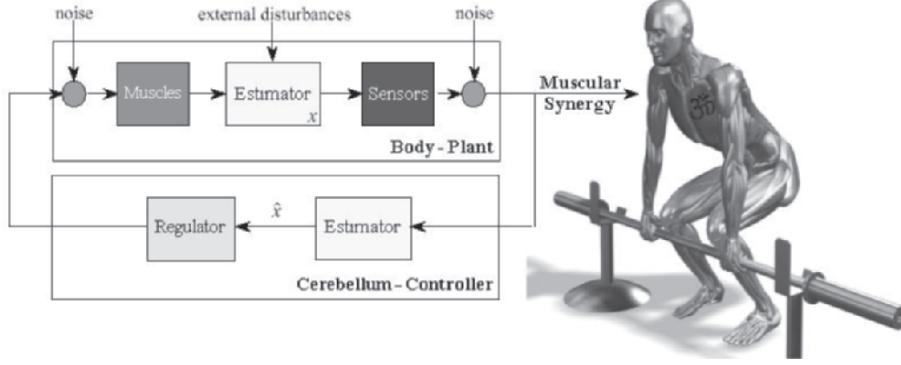


Fig. 3.6. The cerebellum as a motor controller.

organization reveals salient features of cerebellar information processing, but also highlights the importance of systems level analysis for a fuller understanding of the neural mechanisms that underlie behavior.

Hamiltonian Action and Neural Path Integral

Here, we present a *quantum-like adaptive control* approach to modeling the ‘cerebellar mystery’. Corresponding to the affine Hamiltonian control function (3.452) we define the *affine Hamiltonian control action*,

$$S_{aff}[q, p] = \int_{t_{in}}^{t_{out}} d\tau [p_i \dot{q}^i - H_{aff}(q, p)]. \quad (3.454)$$

From the affine Hamiltonian action (3.454) we further derive the associated expression for the *neural phase-space path integral* (in normal units), representing the *cerebellar sensory-motor amplitude* $\langle out|in \rangle$,

$$\begin{aligned} \langle q_{out}^i, p_i^{out} | q_{in}^i, p_i^{in} \rangle &= \int \mathcal{D}[w, q, p] e^{i S_{aff}[q, p]} \\ &= \int \mathcal{D}[w, q, p] \exp \left\{ i \int_{t_{in}}^{t_{out}} d\tau [p_i \dot{q}^i - H_{aff}(q, p)] \right\}, \\ \text{with } \int \mathcal{D}[w, q, p] &= \int \prod_{\tau=1}^n \frac{w^i(\tau) dp_i(\tau) dq^i(\tau)}{2\pi}, \end{aligned} \quad (3.455)$$

where $w_i = w_i(t)$ denote the cerebellar synaptic weights positioned along its neural pathways, being continuously updated using the Hebbian-like self-organizing learning rule (3.451). Given the transition amplitude $\langle out|in \rangle$ (3.455), the *cerebellar sensory-motor transition probability* is defined as its absolute square, $|\langle out|in \rangle|^2$.

In (3.455), $q_{in}^i = q_{in}^i(t)$, $q_{out}^i = q_{out}^i(t)$; $p_i^{in} = p_i^{in}(t)$, $p_i^{out} = p_i^{out}(t)$; $t_{in} \leq t \leq t_{out}$, for all discrete time steps, $t = 1, \dots, n \rightarrow \infty$, and we are allowing for the affine Hamiltonian $H_{aff}(q, p)$ to depend upon all the ($M \leq N$) EMA-angles and angular momenta collectively. Here, we actually systematically took a discretized differential time limit of the form $t_\sigma - t_{\sigma-1} \equiv d\tau$ (both σ and τ denote discrete time steps) and wrote $\frac{(q_\sigma^i - q_{\sigma-1}^i)}{(t_\sigma - t_{\sigma-1})} \equiv \dot{q}^i$. For technical details regarding the path integral calculations on Riemannian and symplectic manifolds (including the standard regularization procedures), see [Kla97, Kla00].

Now, motor learning occurring in the cerebellum can be observed using functional MR imaging, showing changes in the cerebellar action potential, related to the motor tasks (see, e.g., [MA02]). To account for these electrophysiological currents, we need to add the *source* term $J_i(t)q^i(t)$ to the affine Hamiltonian action (3.454), (the current $J_i = J_i(t)$ acts as a source $J_i A^i$ of the *cerebellar electrical potential* $A^i = A^i(t)$),

$$S_{aff}[q, p, J] = \int_{t_{in}}^{t_{out}} d\tau [p_i \dot{q}^i - H_{aff}(q, p) + J_i q^i],$$

which, subsequently gives the cerebellar path integral with the action potential source, coming either from the motor cortex or from other subcortical areas.

Note that the standard *Wick rotation*: $t \mapsto it$ (see [Kla97, Kla00]), makes all our path integrals real, i.e.,

$$\int \mathcal{D}[w, q, p] e^{i S_{aff}[q, p]} \xrightarrow{\text{Wick}} \int \mathcal{D}[w, q, p] e^{-S_{aff}[q, p]},$$

while their subsequent discretization gives the standard thermodynamic *partition functions*,

$$Z = \sum_j e^{-w_j E^j / T}, \quad (3.456)$$

where E^j is the energy eigenvalue corresponding to the affine Hamiltonian $H_{aff}(q, p)$, T is the temperature-like environmental control parameter, and the sum runs over all energy eigenstates (labeled by the index j). From (3.456), we can further calculate all statistical and thermodynamic system properties (see [Fey72]), as for example, *transition entropy* $S = k_B \ln Z$, etc.

3.5.5 Brain-Like Control Functor in Biomechanics

In this final section we present our recent model [IB05] of the complete biomechanical *brain-like control functor*. This is a neurodynamical reflection on our *covariant force law*, $F_i = mg_{ij}a^j$, and its associated *covariant force functor* $\mathcal{F}_* : TT^*M \rightarrow TTM$.

Recall that traditional hierarchical robot control (see [II05]) consists of three levels: the *executive* control-level (at the bottom) performs tracking of

nominal trajectories in internal-joint coordinates, the *strategic* control-level (at the top) performs ‘planning’ of trajectories of an end-effector in external-Cartesian coordinates, and the *tactical* control-level (in the middle) connects other two levels by means of inverse kinematics.

The modern version of the hierarchical robot control includes decision-making done by the neural (or, neuro-fuzzy) classifier to adapt the (manipulator) control to dynamically changing environment.

On the other hand, the so-called ‘intelligent’ approach to robot control typically represents a form of function approximation, which is itself based on some combination of neuro-fuzzy-genetic computations. Many special issues and workshops focusing on physiological models for robot control reflect the increased attention for the development of *cerebellar models* [Sma99, SA98, Sch99, Sch98, Arb98] for learning robot control with functional decomposition, where the main result could be formulated as: *the cerebellum is more than just the function approximator*.

In this section we try to fit between these three approaches for humanoid control, emphasizing the role of muscle-like robot actuators. We present a new, physiologically based, tensor-invariant, hierarchical force control (FC, for short) for the physiologically realistic biomechanics. We consider the muscular torque one-forms F_i as the most important component of human-like motion; therefore we present the sophisticated hierarchical system for the subtle F_i -control: corresponding to the spinal, the cerebellar and cortical levels of human motor control. F_i are first set-up as testing input-signals to biomechanics, and then covariantly updated as feedback 1-forms u_i on each FC-level. On the spinal FC-level the nominal joint-trajectory tracking is proposed in the form of affine Hamiltonian control; here the driving torques are given corrections by spinal-reflex controls. On the cerebellar FC-level, the relation is established between canonical joint coordinates q^i , p_i and gradient *neural-image coordinates* x^i , y_i , representing bidirectional, self-organized, associative memory machine; here the driving torques are given the cerebellar corrections. On the cortical FC-level the topological ‘hyper-joystick’ is proposed as the central FC command-space, selector, with the fuzzy-logic feedback-control map defined on it, giving the cortical corrections to the driving torques.

The model of the spinal FC-level formulated here resembles *autogenetic motor servo*, acting on the spinal-reflex level of the human locomotor control. The model of the cerebellar FC-level formulated here mimics the self-organizing, associative function of the excitatory granule cells and the inhibitory Purkinje cells of the cerebellum [HBB96]. The model of the cortical FC-level presented in this section mimics the synergistic *regulation of locomotor conditioned reflexes* by the cerebellum [HBB96].

We believe that (already mentioned) extremely high order of the driving force redundancy in biomechanics justifies the formulation of the three-level force control system. Also, both brain-like control systems can be easily extended to give $SE(3)$ -based force control for moving inverse kinematics (IK) chains of legs and arms.

Functor Control Machine

In this subsection we define the functor control-machine for the learning control with functional decomposition, by a two-step generalization of the Kalman's theory of linear MIMO-feedback systems. The first generalization puts the Kalman's theory into the pair of mutually dual linear categories **Vect** and **Vect**^{*} of vector spaces and linear operators, with a 'loop-functor' representing the closed-loop control, thus formulating the unique, categorical formalism valid both for the discrete and continual MIMO-systems.

We start with the unique, *feedforward continual-sequential state equation*

$$\dot{x}(t+1) = Ax(t) + Bu(t), \quad y(t) = Cx(t), \quad (3.457)$$

where the nD vector spaces of *state* $X \ni x$, *input* $U \ni u$, and *output* $Y \ni y$ have the corresponding linear operators, respectively $A : X \rightarrow X$, $B : U \rightarrow X$, and $C : X \rightarrow Y$. The modular system theory comprises the *system dynamics*, given by a pair (X, A) , together with a *reachability map* $e : U \rightarrow X$ of the pair (B, A) , and an *observability map* $m : X \rightarrow Y$ of the pair (A, C) . If the reachability map e is surjection the system dynamics (X, A) is called *reachable*; if the observability map m is injection the system dynamics (X, A) is called *observable*. If the system dynamics (X, A) is both reachable and observable, a *composition* $r = m \circ e : U \rightarrow Y$ defines the *total system's response*, which is given by solution of equation (3.457). If the unique solution to the continual-sequential state equation exists, it gives the answer to the (minimal) *realization problem*: find the system S that realizes the given response $r = m \circ e : U \rightarrow Y$ (in the smallest number of discrete states and in the shortest time).

The inverse map $r^{-1} = e^{-1} \circ m^{-1} : Y \rightarrow U$ of the total system's response $r : U \rightarrow Y$ defines the linear *feedback operator* $K : Y \rightarrow U$, given by standard feedback equation

$$u(t) = Ky(t). \quad (3.458)$$

In categorical language, the feedforward system dynamics in the category **Vect** is a pair (X, A) , where $X \in \text{Ob}(\text{Vect})$ is an object in **Vect** and $A : X \rightarrow X \in \text{Mor}(\text{Vect})$ is a **Vect**-morphism. A *feedforward decomposable system* in **Vect** is such a sextuple $S \equiv (X, A, U, B, Y, C)$ that (X, A) is the system dynamics in **Vect**, a **Vect**-morphism $B : U \rightarrow X$ is an *input map*, and a **Vect**-morphism $C : X \rightarrow Y$ is an *output map*. Any object in **Vect** is characterized by mutually dual notions of its *degree* (a number of its input morphisms) and its *codegree* (a number of its output morphisms). Similarly, any decomposable system S in **Vect** has a *reachability map* given by an epimorphism $e = A \circ B : U \rightarrow X$ and its dual *observability map* given by a monomorphism $m = C \circ A : X \rightarrow Y$; their composition $r = m \circ e : U \rightarrow Y$ in **Mor**(**Vect**) defines the total system's response in **Vect** given by the unique solution of the continual-sequential state equation (3.457) [IS01].

The dual of the total system's response, defined by the feedback equation (3.458), is the *feedback morphism* $K = e^{-1} \circ m^{-1} : Y \rightarrow U$ belonging to the dual category $\mathbf{Vect}^*$.

In this way, the linear, closed-loop, continual-sequential MIMO-system (3.457–3.458) represents the *linear iterative loop functor* $\mathcal{L} : \mathbf{Vect} \Rightarrow \mathbf{Vect}^*$.

Our second generalization represents a *natural system process* $\Xi[\mathcal{L}]$, that transforms the linear loop functor $\mathcal{L} : \mathbf{Vect} \Rightarrow \mathbf{Vect}^*$ – into the *nonlinear loop functor* $\mathcal{NL} : \mathcal{CAT} \Rightarrow \mathcal{CAT}^*$ – between two mutually dual nonlinear categories $\mathcal{CAT}$ and $\mathcal{CAT}^*$. We apply the natural process Ξ , separately

1. To the feedforward decomposable system
 $S \equiv (X, A, U, B, Y, C)$ in $\mathbf{Vect}$, and
2. To the feedback morphism $K = e^{-1} \circ m^{-1} : Y \rightarrow U$ in $\mathbf{Vect}^*$.

Under the action of the natural process Ξ , the linear feedforward system dynamics (X, A) in $\mathbf{Vect}$ transforms into a nonlinear feedforward Ξ -dynamics $(\Xi[X], \Xi[A])$ in $\mathcal{CAT}$, represented by a *nonlinear feedforward decomposable system*, $\Xi[S] \equiv (\Xi[X], \Xi[A], \Xi[U], \Xi[B], \Xi[Y], \Xi[C])$.

The reachability map transforms into the *input process* $\Xi[e] = \Xi[A] \circ \Xi[B] : \Xi[U] \rightarrow \Xi[X]$, while its dual, observability map transforms into the *output process* $\Xi[m] = \Xi[C] \circ \Xi[A] : \Xi[X] \rightarrow \Xi[Y]$. In this way the total response of the linear system $r = m \circ e : U \rightarrow Y$ in $\mathbf{Mor}(\mathbf{Vect})$ transforms into the *nonlinear system behavior*, $\Xi[r] = \Xi[m] \circ \Xi[e] : \Xi[U] \rightarrow \Xi[Y]$ in $\mathbf{Mor}(\mathcal{CAT})$. Obviously, $\Xi[r]$, if exists, is given by a nonlinear Ξ -transform of the linear state equations (3.457–3.458).

Analogously, the linear feedback morphism $K = e^{-1} \circ m^{-1} : Y \rightarrow U$ in $\mathbf{Mor}(\mathbf{Vect}^*)$ transforms into the nonlinear feedback morphism $\Xi[K] = \Xi[e^{-1}] \circ \Xi[m^{-1}] : \Xi[Y] \rightarrow \Xi[U]$ in $\mathbf{Mor}(\mathcal{CAT}^*)$.

In this way, the natural system process $\Xi : \mathcal{L} \Rightarrow \mathcal{NL}$ is established. That means that the nonlinear loop functor $L = \Xi[\mathcal{L}] : \mathcal{CAT} \Rightarrow \mathcal{CAT}^*$ is defined out of the linear, closed-loop, continual-sequential MIMO-system (3.457).

In this section we formulate the nonlinear loop functor $L = \Xi[\mathcal{L}] : \mathcal{CAT} \Rightarrow \mathcal{CAT}^*$ for various hierarchical levels of muscular-like FC.

Spinal Control Level

Our first task is to establish the nonlinear loop functor $L = \Xi[\mathcal{L}] : \mathcal{EX} \Rightarrow \mathcal{EX}^*$ on the category $\mathcal{EX}$ of spinal FC-level.

Recall that our dissipative, driven δ -Hamiltonian biomechanical system on the configuration manifold M is, in local canonical-symplectic coordinates $q^i, p_i \in U_p$ on the momentum phase-space manifold T^*M , given by autonomous equations

$$\dot{q}^i = \frac{\partial H_0}{\partial p_i} + \frac{\partial R}{\partial p_i}, \quad (i = 1, \dots, N) \quad (3.459)$$

$$\dot{p}_i = F_i - \frac{\partial H_0}{\partial q^i} + \frac{\partial R}{\partial q^i}, \quad (3.460)$$

$$q^i(0) = q_0^i, \quad p_i(0) = p_i^0, \quad (3.461)$$

including contravariant equation (3.459) – the velocity vector-field, and covariant equation (3.460) – the force 1-form, together with initial joint angles q_0^i and momenta p_i^0 . Here the physical Hamiltonian function $H_0 : T^*M \rightarrow \mathbb{R}$ represents the total biomechanical energy function, in local canonical coordinates $q^i, p_i \in U_p$ on T^*M given by

$$H_0(q, p) = \frac{1}{2} g^{ij} p_i p_j + V(q),$$

where $g^{ij} = g^{ij}(q, m)$ denotes the contravariant material metric tensor.

Now, the *control Hamiltonian function* $H_\gamma : T^*M \rightarrow \mathbb{R}$ of FC is in local canonical coordinates on T^*M defined by [NS90]

$$H_\gamma(q, p, u) = H_0(q, p) - q^i u_i, \quad (i = 1, \dots, N) \quad (3.462)$$

where $u_i = u_i(t, q, p)$ are feedback-control 1-forms, representing the spinal FC-level u -corrections to the covariant torques $F_i = F_i(t, q, p)$.

Using δ -Hamiltonian biomechanical system (3.459–3.461) and the control Hamiltonian function (3.462), control γ_δ -Hamiltonian FC-system can be defined as

$$\begin{aligned} \dot{q}^i &= \frac{\partial H_\gamma(q, p, u)}{\partial p_i} + \frac{\partial R(q, p)}{\partial p_i}, \\ \dot{p}_i &= F_i - \frac{\partial H_\gamma(q, p, u)}{\partial q^i} + \frac{\partial R(q, p)}{\partial q^i}, \\ o^i &= -\frac{\partial H_\gamma(q, p, u)}{\partial u_i}, \quad (i = 1, \dots, N) \\ q^i(0) &= q_0^i, \quad p_i(0) = p_i^0, \end{aligned}$$

where $o^i = o^i(t)$ represent FC natural outputs which can be different from commonly used joint angles.

If nominal reference outputs $o_R^i = o_R^i(t)$ are known, the simple PD *stiffness-servo* [Whi87] could be formulated, via *error function* $e(t) = o^j - o_R^j$, in covariant form

$$u_i = K_o \delta_{ij} (o^j - o_R^j) + K_{\dot{o}} \delta_{ij} (\dot{o}^j - \dot{o}_R^j), \quad (3.463)$$

where K s are the control-gains and δ_{ij} is the Kronecker tensor.

If natural outputs o^i *actually* are the joint angles and nominal canonical trajectories ($q_R^i = q_R^i(t)$, $p_i^R = p_i^R(t)$) are known, then the stiffness-servo (3.463) could be formulated in canonical form as

$$u_i = K_q \delta_{ij} (q^i - q_R^i) + K_p (p_i - p_i^R).$$

Now, using the fuzzified μ –Hamiltonian biomechanical system with fuzzy system numbers (i.e, imprecise segment lengths, masses and moments of inertia, joint dampings and muscular actuator parameters)

$$\dot{q}^i = \frac{\partial H_0(q, p, \sigma_\mu)}{\partial p_i} + \frac{\partial R}{\partial p_i}, \quad (3.464)$$

$$\dot{p}_i = \bar{F}_i - \frac{\partial H_0(q, p, \sigma_\mu)}{\partial q^i} + \frac{\partial R}{\partial q^i}, \quad (3.465)$$

$$q^i(0) = \bar{q}_0^i, \quad p_i(0) = \bar{p}_i^0, \quad (i = 1, \dots, N), \quad (3.466)$$

and the control Hamiltonian function (3.462), γ_μ –Hamiltonian FC–system can be defined as

$$\begin{aligned} \dot{q}^i &= \frac{\partial H_\gamma(q, p, u, \sigma_\mu)}{\partial p_i} + \frac{\partial R(q, p)}{\partial p_i}, \\ \dot{p}_i &= \bar{F}_i - \frac{\partial H_\gamma(q, p, u, \sigma_\mu)}{\partial q^i} + \frac{\partial R(q, p)}{\partial q^i}, \\ \dot{\sigma}^i &= -\frac{\partial H_\gamma(q, p, u, \sigma_\mu)}{\partial u_i}, \quad q^i(0) = \bar{q}_0^i, \quad p_i(0) = \bar{p}_i^0, \end{aligned}$$

where $\bar{\sigma}^i = \bar{\sigma}^i(t)$ represent the fuzzified natural outputs.

Finally, applying stochastic forces (diffusion fluctuations $B_{ij}[q^i(t), t]$ and discontinuous jumps in the form of ND Wiener process $W^j(t)$), i.e., using the fuzzy–stochastic $[\mu\sigma]$ –Hamiltonian biomechanical system

$$dq^i = \left(\frac{\partial H_0(q, p, \sigma_\mu)}{\partial p_i} + \frac{\partial R}{\partial p_i} \right) dt, \quad (3.467)$$

$$\begin{aligned} dp_i &= B_{ij}[q^i(t), t] dW^j(t) + \\ &\quad \left(\bar{F}_i - \frac{\partial H_0(q, p, \sigma_\mu)}{\partial q^i} + \frac{\partial R}{\partial q^i} \right) dt, \end{aligned} \quad (3.468)$$

$$q^i(0) = \bar{q}_0^i, \quad p_i(0) = \bar{p}_i^0, \quad (3.469)$$

and the control Hamiltonian function (3.462), $\gamma_{\mu\sigma}$ –Hamiltonian FC–system can be defined as

$$\begin{aligned} dq^i &= \left(\frac{\partial H_\gamma(q, p, u, \sigma_\mu)}{\partial p_i} + \frac{\partial R(q, p)}{\partial p_i} \right) dt, \\ dp_i &= B_{ij}[q^i(t), t] dW^j(t) + \\ &\quad \left(\bar{F}_i - \frac{\partial H_\gamma(q, p, u, \sigma_\mu)}{\partial q^i} + \frac{\partial R(q, p)}{\partial q^i} \right) dt, \\ d\bar{\sigma}^i &= -\frac{\partial H_\gamma(q, p, u, \sigma_\mu)}{\partial u_i} dt, \quad (i = 1, \dots, N) \\ q^i(0) &= \bar{q}_0^i, \quad p_i(0) = \bar{p}_i^0. \end{aligned}$$

If we have the case that not all of the configuration joints on the configuration manifold M are active in the specified robot task, we can introduce the coupling Hamiltonians $H^j = H^j(q, p)$, $j = 1, \dots, M \leq N$, corresponding to the system's active joints, and we come to *affine Hamiltonian function* $H_a : T^*M \rightarrow \mathbb{R}$, in local canonical coordinates on T^*M given as [NS90]

$$H_a(q, p, u) = H_0(q, p) - H^j(q, p) u_j. \quad (3.470)$$

Using δ -Hamiltonian biomechanical system (3.459–3.461) and the affine Hamiltonian function (3.470), affine a_δ -Hamiltonian FC-system can be defined as

$$\dot{q}^i = \frac{\partial H_0(q, p)}{\partial p_i} - \frac{\partial H^j(q, p)}{\partial p_i} u_j + \frac{\partial R}{\partial p_i}, \quad (3.471)$$

$$\dot{p}_i = F_i - \frac{\partial H_0(q, p)}{\partial q^i} + \frac{\partial H^j(q, p)}{\partial q^i} u_j + \frac{\partial R}{\partial q^i}, \quad (3.472)$$

$$o^i = -\frac{\partial H_a(q, p, u)}{\partial u_i} = H^j(q, p), \quad (3.473)$$

$$q^i(0) = q_0^i, \quad p_i(0) = p_i^0, \quad (3.474)$$

$$(i = 1, \dots, N; \quad j = 1, \dots, M \leq N).$$

Using the Lie-derivative exact feedback linearization (see (2.2.2) above), and applying the *constant relative degree* r (see [Isi89, SI89]) to all N joints of the affine a_δ -Hamiltonian FC-system (3.471–3.474), the control law for asymptotic tracking the reference outputs o_R^j could be formulated as

$$u_j = \frac{\dot{o}_R^{(r)j} - \mathcal{L}_f^{(r)} H^j + \sum_{s=1}^r \gamma_{s-1} (o_R^{(s-1)j} - \mathcal{L}_f^{(s-1)} H^j)}{\mathcal{L}_g \mathcal{L}_f^{(r-1)} H^j},$$

where standard MIMO-vector-fields f and g are given by

$$f = \left(\frac{\partial H_0}{\partial p_i}, -\frac{\partial H_0}{\partial q^i} \right), \quad g = \left(-\frac{\partial H^j}{\partial p_i}, \frac{\partial H^j}{\partial q^i} \right)$$

and γ_{s-1} are the coefficients of linear differential equation of order r for the error function $e(t) = o^j - o_R^j$

$$e^{(r)} + \gamma_{r-1} e^{(r-1)} + \dots + \gamma_1 e^{(1)} + \gamma_0 e = 0.$$

Using the fuzzified μ -Hamiltonian biomechanical system (3.464–3.466) and the affine Hamiltonian function (3.470), affine a_μ -Hamiltonian FC-system can be defined as

$$\begin{aligned}
\dot{q}^i &= \frac{\partial H_0(q, p, \sigma_\mu)}{\partial p_i} - \frac{\partial H^j(q, p, \sigma_\mu)}{\partial p_i} u_j + \frac{\partial R(q, p)}{\partial p_i}, \\
\dot{p}_i &= \bar{F}_i - \frac{\partial H_0(q, p, \sigma_\mu)}{\partial q^i} + \frac{\partial H^j(q, p, \sigma_\mu)}{\partial q^i} u_j + \frac{\partial R(q, p)}{\partial q^i}, \\
\dot{\sigma}^i &= -\frac{\partial H_a(q, p, u, \sigma_\mu)}{\partial u_i} = H^j(q, p, \sigma_\mu), \\
q^i(0) &= \bar{q}_0^i, \quad p_i(0) = \bar{p}_i^0, \quad (i = 1, \dots, N; j = 1, \dots, M \leq N).
\end{aligned}$$

Using the fuzzy-stochastic $[\mu\sigma]$ -Hamiltonian biomechanical system (3.467–3.469) and the affine Hamiltonian function (3.470), affine $a_{\mu\sigma}$ -Hamiltonian FC-system can be defined as

$$\begin{aligned}
dq^i &= \left(\frac{\partial H_0(q, p, \sigma_\mu)}{\partial p_i} - \frac{\partial H^j(q, p, \sigma_\mu)}{\partial p_i} u_j + \frac{\partial R(q, p)}{\partial p_i} \right) dt, \\
dp_i &= B_{ij}[q^i(t), t] dW^j(t) + \\
&\quad \left(\bar{F}_i - \frac{\partial H_0(q, p, \sigma_\mu)}{\partial q^i} + \frac{\partial H^j(q, p, \sigma_\mu)}{\partial q^i} u_j + \frac{\partial R(q, p)}{\partial q^i} \right) dt, \\
d\sigma^i &= -\frac{\partial H_a(q, p, u, \sigma_\mu)}{\partial u_i} dt = H^j(q, p, \sigma_\mu) dt, \\
q^i(0) &= \bar{q}_0^i, \quad p_i(0) = \bar{p}_i^0, \quad (i = 1, \dots, N; j = 1, \dots, M \leq N).
\end{aligned}$$

Being high-degree and highly nonlinear, all of these affine control systems are extremely sensitive upon the variation of parameters, inputs, and initial conditions. The sensitivity function S of the affine Hamiltonian $H_a(q, p, u)$ upon the parameters β_i (representing segment lengths L_i , masses m_i , moments of inertia J_i and joint dampings b_i , see [IS01, II06a]), is in the case of a_δ -Hamiltonian FC-system defined as

$$S(H, \beta) = \frac{\beta_i}{H_a(q, p, u)} \frac{\partial H_a(q, p, u)}{\partial \beta_i},$$

and similarly in other two a_μ - and $a_{\mu\sigma}$ -cases.

The three affine FC-level systems a_δ , a_μ and $a_{\mu\sigma}$, resemble (in a fuzzy-stochastic-Hamiltonian form), Houk's autogenetic motor servo of muscle spindle and Golgi tendon proprioceptors [Hou79], correcting the covariant driving torques $F_i = F_i(t, q, p)$ by local 'reflex controls' $u_i(t, q, p)$. They form the nonlinear loop functor $L = \Xi[\mathcal{L}] : \mathcal{E}\mathcal{X} \Rightarrow \mathcal{E}\mathcal{X}^*$.

Cerebellar Control Level

Our second task is to establish the nonlinear loop functor $L = \Xi[\mathcal{L}] : \mathcal{TA} \Rightarrow \mathcal{TA}^*$ on the category $\mathcal{TA}$ of the cerebellar FC-level. Here we present an oscillatory neurodynamical (x, y, ω) -system (adapted from [IJB99a]), a bidirectional, self-organized, associative-memory machine, resembling the function

of a set of excitatory granule cells and inhibitory Purkinje cells in the middle layer of the cerebellum (see [EIS67, HBB96]). The neurodynamical (x, y, ω) -system acts on *neural-image manifold* M_{im}^N of the configuration manifold M^N as a pair of smooth, ‘1 – 1’ and ‘onto’ maps (Ψ, Ψ^{-1}) , where $\Psi : M^N \rightarrow M_{im}^N$ represents the feedforward map, and $\Psi^{-1} : M_{im}^N \rightarrow M^N$ represents the feedback map. Locally, it is defined in Riemannian neural coordinates $x^i, y_i \in V_y$ on M_{im}^N , which are in bijective correspondence with symplectic joint coordinates $q^i, p_i \in U_p$ on T^*M .

The (x, y, ω) -system is formed out of two distinct, yet nonlinearly-coupled neural subsystems, with $A^i(q)$ (3.477) and $B_i(p)$ (3.478) as system inputs, and the feedback-control 1-forms u_i (3.483) as system outputs:

1. Granule cells excitatory (contravariant) and Purkinje cells inhibitory (covariant) activation (x, y) -dynamics (3.475–3.478), defined respectively by a vector-field $\dot{x}^i = x^i(t) : M \rightarrow TM$, representing a cross-section of the tangent bundle TM , and a 1-form $y_i = y_i(t) : M \rightarrow T^*M$, representing a cross-section of the cotangent bundle T^*M ; and
2. Excitatory and inhibitory unsupervised learning (ω) -dynamics (3.478–3.480) generated by random differential Hebbian learning process (3.481–3.483), defined respectively by contravariant synaptic tensor-field $\omega^{ij} = \omega^{ij}(t) : M \rightarrow TTM_{im}^N$ and covariant synaptic tensor-field $\omega_{ij} = \omega_{ij}(t) : M \rightarrow T^*T^*M$, representing cross-sections of contravariant and covariant tensor bundles, respectively.

The system equations are defined as

$$\dot{x}^i = A^i(q) + \omega^{ij} f_j(y) - x^i, \quad (3.475)$$

$$\dot{y}_i = B_i(p) + \omega_{ij} f^j(x) - y_i, \quad (3.476)$$

$$A^i(q) = K_q(q^i - q_R^i), \quad (3.477)$$

$$B_i(p) = K_p(p_i^R - p_i), \quad (3.478)$$

$$\dot{\omega}^{ij} = -\omega^{ij} + I^{ij}(x, y), \quad (3.479)$$

$$\dot{\omega}_{ij} = -\omega_{ij} + I_{ij}(x, y), \quad (3.480)$$

$$I^{ij} = f^i(x) f^j(y) + \dot{f}^i(x) \dot{f}^j(y) + \sigma^{ij}, \quad (3.481)$$

$$I_{ij} = f_i(x) f_j(y) + \dot{f}_i(x) \dot{f}_j(y) + \sigma_{ij}, \quad (3.482)$$

$$u_i = \frac{1}{2}(\delta_{ij} x^i + y_i), \quad (i, j = 1, \dots, N). \quad (3.483)$$

Here ω is a symmetric 2nd order synaptic tensor-field; $I^{ij} = I^{ij}(x, y, \sigma)$ and $I_{ij} = I_{ij}(x, y, \sigma)$ respectively denote contravariant-excitatory and covariant-inhibitory random differential Hebbian innovation-functions with tensorial Gaussian noise σ (in both variances); f s and $\dot{f}$ s denote sigmoid activation functions ($f = \tanh(\cdot)$) and corresponding signal velocities ($\dot{f} = 1 - f^2$), respectively in both variances;

$A^i(q)$ and $B_i(p)$ are contravariant-excitatory and covariant-inhibitory neural inputs to granule and Purkinje cells, respectively; u_i are the corrections to the feedback-control 1-forms on the cerebellar FC-level.

Nonlinear activation (x, y) -dynamics (3.475–3.478), describes a two-phase biological neural oscillator field, in which excitatory neural field excites inhibitory neural field, which itself reciprocally inhibits the excitatory one. (x, y) -dynamics represents a nonlinear extension of a linear, Lyapunov-stable, conservative, gradient system, defined in local neural coordinates $x^i, y_i \in V_y$ on T^*M as

$$\dot{x}^i = -\frac{\partial \Phi}{\partial y_i} = \omega^{ij}y_j - x^i, \quad \dot{y}_i = -\frac{\partial \Phi}{\partial x^i} = \omega_{ij}x^j - y_i. \quad (3.484)$$

The gradient system (3.484) is derived from scalar, neuro-synaptic action potential $\Phi : T^*M \rightarrow \mathbb{R}$, given by a negative, smooth bilinear form in $x^i, y_i \in V_y$ on T^*M as

$$-2\Phi = \omega_{ij}x^ix^j + \omega^{ij}y_iy_j - 2x^iy_i, \quad (i, j = 1, \dots, N), \quad (3.485)$$

which itself represents a Ψ -image of the Riemannian metrics $g : TM \rightarrow \mathbb{R}$ on the configuration manifold M .

The nonlinear oscillatory activation (x, y) -dynamics (3.475–3.478) is get from the linear conservative dynamics (3.484) by adding configuration-dependent inputs A^i and B_i , as well as sigmoid activation functions f_j and f^j , respectively. It represents an interconnected pair of excitatory and inhibitory neural fields.

Both variant-forms of learning (ω) -dynamics (3.479–3.480) are given by generalized unsupervised (self-organizing) Hebbian learning scheme (see [Kos92]) in which $\dot{\omega}_{ij}$ (resp. $\dot{\omega}^{ij}$) denotes the new-update value, $-\omega_{ij}$ (resp. $-\omega^{ij}$) corresponds to the old value and $I_{ij}(x^i, y_j)$ (resp. $I^{ij}(x^i, y_j)$) is the innovation function of the symmetric 2nd order synaptic tensor-field ω . The nonlinear innovation functions I_{ij} and I^{ij} are defined by random differential Hebbian learning process (3.481–3.482). As ω is symmetric and zero-trace coupling synaptic tensor, the conservative linear activation dynamics (3.484) is equivalent to the rule that the state of each neuron (in both neural fields) is changed in time iff the scalar action potential Φ (3.485), is lowered. Therefore, the scalar action potential Φ represents the monotonically decreasing Lyapunov function (such that $\dot{\Phi} \leq 0$) for the conservative linear dynamics (3.484), which converges to a local minimum or ground state of Φ . That is to say, the system (3.484) moves in the direction of decreasing the scalar action potential Φ , and when both $\dot{x}^i = 0$ and $\dot{y}_i = 0$ for all $i = 1, \dots, N$, the steady state is reached.

In this way, the neurodynamical (x, y, ω) -system acts as tensor-invariant self-organizing (excitatory / inhibitory) associative memory machine, resembling the set of granule and Purkinje cells of cerebellum [HBB96].

The feedforward map $\Psi : M \rightarrow M$ is realized by the inputs $A^i(q)$ and $B_i(p)$ to the (x, y, ω) -system, while the feedback map $\Psi^{-1} : M \rightarrow M$ is

realized by the system output, i.e., the feedback-control 1-forms $u_i(x, y)$. These represent the cerebellar FC-level corrections to the covariant torques $F_i = F_i(t, q, p)$.

The tensor-invariant form of the oscillatory neurodynamical (x, y, ω) -system (3.475–3.483) resembles the associative action of the granule and Purkinje cells in the tuning of the limb cortico-rubro-cerebellar recurrent network [HBB96], giving the cerebellar correction $u_i(x, y)$ to the covariant driving torques $F_i = F_i(t, q, p)$. In this way (x, y, ω) -system forms the nonlinear loop functor $L = \Xi[\mathcal{L}] : \mathcal{TA} \Rightarrow \mathcal{TA}^*$.

Cortical Control Level

Our third task is to establish the nonlinear loop functor $L = \Xi[\mathcal{L}] : \mathcal{ST} \Rightarrow \mathcal{ST}^*$ on the category $\mathcal{ST}$ of the cortical FC-level.

Recall that for the purpose of cortical control, the purely rotational biomechanical manifold M could be firstly reduced to N -torus and subsequently transformed to N -cube ('hyper-joystick'), using the following geometrical techniques [II06b].

Denote by S^1 the constrained unit circle in the complex plane. This is an Abelian Lie group. We have two reduction homeomorphisms

$$SO(3) \gtrsim SO(2) \triangleright SO(2) \triangleright SO(2), \quad \text{and} \quad SO(2) \approx S^1,$$

where ' $\triangleright$ ' denotes the noncommutative semidirect product.

Next, let I^N be the unit cube $[0, 1]^N$ in $\mathbb{R}^N$ and ' $\sim$ ' an equivalence relation on $\mathbb{R}^N$ get by 'gluing' together the opposite sides of I^N , preserving their orientation. Therefore, M can be represented as the quotient space of $\mathbb{R}^N$ by the space of the integral lattice points in $\mathbb{R}^N$, that is a constrained torus T^N :

$$\mathbb{R}^N / Z^N = I^N / \sim \cong \prod_{i=1}^N S_i^1 \equiv \{(q^i, i = 1, \dots, N) : \text{mod } 2\pi\} = T^N.$$

In the same way, the momentum phase-space manifold T^*M can be represented by T^*T^N .

Conversely by 'ungluing' the configuration space we get the primary unit cube. Let ' $\sim^*$ ' denote an equivalent decomposition or 'ungluing' relation. By the *Tychonoff product-topology theorem*, for every such quotient space there exists a 'selector' such that their quotient models are homeomorphic, that is, $T^N / \sim^* \approx A^N / \sim^*$. Therefore I_q^N represents a 'selector' for the configuration torus T^N and can be used as an N -directional ' $\hat{q}$ -command-space' for FC. Any subset of DOF on the configuration torus T^N representing the joints included in the general biomechanics has its simple, rectangular image in the rectified $\hat{q}$ -command space – selector I_q^N , and any joint angle q^i has its rectified image $\hat{q}^i$.

In the case of an end-effector, $\hat{q}^i$ reduces to the position vector in external-Cartesian coordinates z^r ($r = 1, \dots, 3$). If orientation of the end-effector can be neglected, this gives a topological solution to the standard inverse kinematics problem.

Analogously, all momenta $\hat{p}_i$ have their images as rectified momenta $\hat{p}_i$ in the $\hat{p}$ -command space – selector I_p^N . Therefore, the total momentum phase-space manifold T^*T^N gets its ‘cortical image’ as the $(\widehat{q}, \widehat{p})$ -command space, a trivial $2ND$ bundle $I_q^N \times I_p^N$.

Now, the simplest way to perform the feedback FC on the cortical $(\widehat{q}, \widehat{p})$ -command space $I_q^N \times I_p^N$, and also to mimic the cortical-like behavior [1,2], is to use the $2ND$ fuzzy-logic controller, in pretty much the same way as in popular ‘inverted pendulum’ examples [Kos92, Kos96].

We present the fuzzy feedback-control map Ξ that maps all the rectified joint angles and momenta into the feedback-control 1-forms

$$\Xi : (\hat{q}^i(t), \hat{p}_i(t)) \mapsto u_i(t, q, p), \quad (3.486)$$

so that their corresponding universes of discourse, $\hat{M}^i = (\hat{q}_{max}^i - \hat{q}_{min}^i)$, $\hat{P}_i = (\hat{p}_i^{max} - \hat{p}_i^{min})$ and $U_i = (u_i^{max} - u_i^{min})$, respectively, are mapped as

$$\Xi : \prod_{i=1}^N \hat{M}^i \times \prod_{i=1}^N \hat{P}_i \rightarrow \prod_{i=1}^N U_i. \quad (3.487)$$

The $2N$ -D map Ξ (3.486–3.487) represents a *fuzzy inference system*, defined by (adapted from [IJB99b]):

1. *Fuzzification* of the crisp *rectified* and *discretized* angles, momenta and controls using Gaussian-bell membership functions

$$\mu_k(\chi) = \exp\left[-\frac{(\chi - m_k)^2}{2\sigma_k}\right], \quad (k = 1, 2, \dots, 9),$$

where $\chi \in D$ is the common symbol for $\hat{q}^i$, $\hat{p}_i$ and $u_i(q, p)$ and D is the common symbol for M^i , $\hat{P}_i$ and i ; the mean values m_k of the seven partitions of each universe of discourse D are defined as $m_k = \lambda_k D + \chi_{min}$, with partition coefficients λ_k uniformly spanning the range of D , corresponding to the set of nine linguistic variables $L = \{NL, NB, NM, NS, ZE, PS, PM, PB, PL\}$; standard deviations are kept constant $\sigma_k = D/9$. Using the linguistic vector L , the 9×9 FAM (fuzzy associative memory) matrix (a ‘linguistic phase-plane’), is heuristically defined for each human joint, in a symmetrical weighted form

$$\mu_{kl} = \varpi_{kl} \exp\{-50[\lambda_k + u(q, p)]^2\}, \quad (k, l = 1, 2, \dots, 9)$$

with weights $\varpi_{kl} \in \{0.6, 0.6, 0.7, 0.7, 0.8, 0.8, 0.9, 0.9, 1.0\}$.

2. *Mamdani inference* is used on each FAM-matrix μ_{kl} for all human joints: (i) $\mu(\hat{q}^i)$ and $\mu(\hat{p}_i)$ are combined inside the fuzzy IF-THEN rules using AND (Intersection, or Minimum) operator,

$$\mu_k[\bar{u}_i(q, p)] = \min_l \{\mu_{kl}(\hat{q}^i), \mu_{kl}(\hat{p}_i)\}.$$

- (ii) the output sets from different IF-THEN rules are then combined using OR (Union, or Maximum) operator, to get the final output, fuzzy-covariant torques,

$$\mu[u_i(q, p)] = \max_k \{\mu_k[\bar{u}_i(q, p)]\}.$$

3. *Defuzzification* of the fuzzy controls $\mu[u_i(q, p)]$ with the ‘center of gravity’ method

$$u_i(q, p) = \frac{\int \mu[u_i(q, p)] du_i}{\int du_i},$$

to update the crisp feedback-control 1-forms $u_i = u_i(t, q, p)$. These represent the cortical FC-level corrections to the covariant torques $F_i = F_i(t, q, p)$.

Operationally, the construction of the cortical $(\widehat{q, p})$ -command space $I_q^N \times I_p^N$ and the 2ND feedback map Ξ (3.486–3.487), mimic the regulation of locomotor conditioned reflexes by the motor cortex [HBB96], giving the cortical correction to the covariant driving torques F_i . Together they form the nonlinear loop functor $\mathcal{NL} = \Xi[\mathcal{L}] : \mathcal{ST} \Rightarrow \mathcal{ST}^*$.

A sample output from the leading human-motion simulator, *Human Biodynamics Engine* (developed by the authors in Defence Science & Technology Organisation, Australia), is given in Figure 3.7, giving the sophisticated 264 DOF analysis of adult male running with the speed of 5 m/s.

3.5.6 Concurrent and Weak Functorial Machines

In this section we first present a concept of machine concurrency, as a modern development of parallel and distributed information processing realized in ANNs, and after that an abstract concept of functorial machine.

Concurrent Machines and Higher-Dimensional Automata

A *sequential machine* (i.e., an ordinary state-machine without concurrency) is a set of states, also called 0-transitions, and a set of 1-transitions from a given state to another one.

On the other hand, a *concurrent machine* consist of a set of states and a set of 1-transitions, but has also the capability of carrying out several 1-transitions at the same time. A concurrent machine is a computer in which

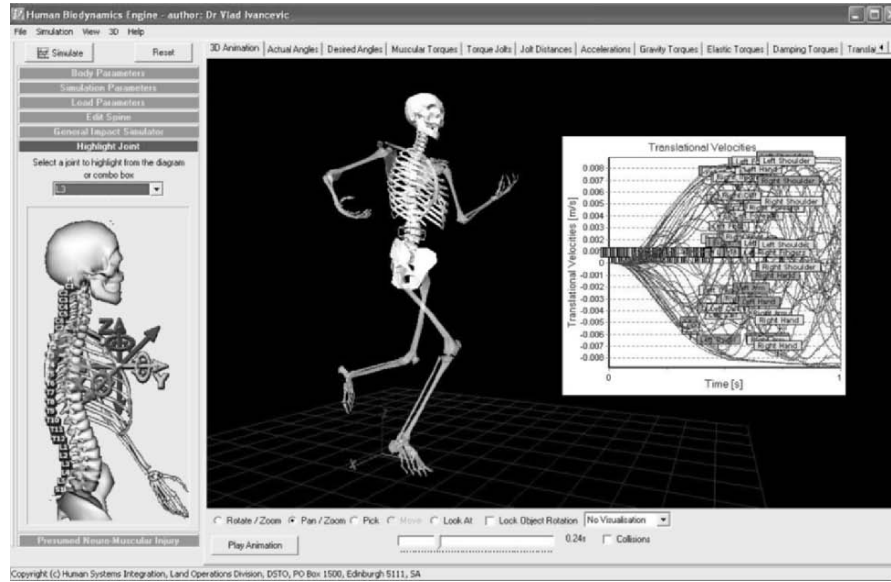


Fig. 3.7. Sample output from the Human Biodynamics Engine: running with the speed of 5 m/s.

several tasks can be performed at the same time. This can be a true parallelism like in the case of several processors running concurrently, or a mono-processor machine in which a unique CPU shares its time between several different tasks.

The *homotopical analysis of concurrency* consists of dealing with concurrent machines using the tools and methods of algebraic topology by working with execution paths up to homotopy [Gou95, FGR98]. The geometric model of concurrent processes allows to formalize very precisely the notion of dihomotopy. Two concurrent processes (i.e., the corresponding geometric models) are said dihomotopic when they can be geometrically continuously deformed to each other without changing their computer-scientific properties (see [Gau02a, Gau02b, Gau02c, GG03, Gau03]).

Consider a system with finitely many processes running altogether. We assume that each process starts at (local time) 0 and finishes at (local time) 1; the P and V actions correspond to sequences of real numbers between 0 and 1, which reflect the order of the P 's and V 's. The initial state is $(0, \dots, 0)$ and the final state is $(1, \dots, 1)$.

The basic idea is to give a description of what can happen when several processes are modifying shared resources. Given a shared resource a , we see it as its associated semaphore that rules its behavior with respect to processes. For example, if a is an ordinary shared variable, it is customary to use its semaphore to ensure that only one process at a time can write on it (this is mutual exclusion). A semaphore is nothing but a register which counts the

number of times a shared object can still be accessed by processes. In the case of usual shared variables, this register is initialized with value 1, processes trying to access (read or write) on the corresponding variable compete in order to get it first, then the semaphore value is decreased: we say that the semaphore has been locked by the process. When it is equal to zero, all processes trying to access this semaphore are blocked, waiting for the process which holds the lock to relinquish it, typically when it has finished reading or writing on the corresponding variable: the value of the semaphore is then increased.

For example, when the semaphores are initialized with value one, meaning that they are associated with shared variables accessed in a mutually exclusive manner, they are called binary semaphores. When a shared data (identified with its semaphore) can be accessed by one or more processes, meaning that the corresponding semaphore has been initialized with a value greater than one, it is called a counting semaphore.

Given n deterministic sequential processes $Q_1, \dots, Q_n$, abstracted as a sequence of locks and unlocks on (semaphores associated with) shared objects, $Q_i = R^1 a_i^1 . R^2 a_i^2 \dots R^{n_i} a_i^{n_i}$ (R^k being P or V , there is a natural way to understand the possible behaviors of their concurrent execution, by associating to each process a coordinate line in $\mathbb{R}^n$. The state of the system corresponds to a point in $\mathbb{R}^n$, whose i th coordinate describes the state (or ‘local time’) of the i th processor.

Consider a system with finitely many processes running altogether. We assume that each process starts at (local time) 0 and finishes at (local time) 1; the P and V actions correspond to sequences of real numbers between 0 and 1, which reflect the order of the P 's and V 's. The initial state is $(0, \dots, 0)$ and the final state is $(1, \dots, 1)$. An example consisting of the two processes $T_1 = Pa.Pb.Vb.Va$ and $T_2 = Pb.Pa.Va.Vb$ induces the 2D *progress graph* of Figure 3.8.

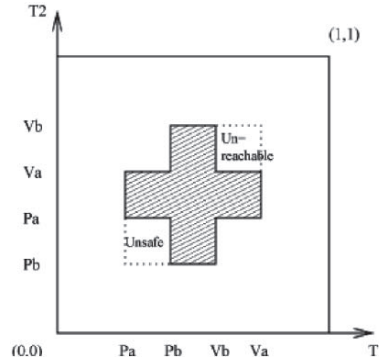


Fig. 3.8. Example of a *progress graph* (adapted from [Gau02a]).

The shaded area represents states which are not allowed in any execution path, since they correspond to mutual exclusion. Such states constitute the *forbidden area*. An *execution path* is a path from the initial state $(0, \dots, 0)$ to the final state $(1, \dots, 1)$ avoiding the forbidden area and increasing in each coordinate – time cannot run backwards. We call these paths *directed paths* or *dipaths*. This entails that paths reaching the states in the dashed square underneath the forbidden region, marked ‘unsafe’ are deemed to deadlock, i.e., they cannot possibly reach the allowed terminal state which is $(1, 1)$ here. Similarly, by reversing the direction of time, the states in the square above the forbidden region, marked ‘unreachable’, cannot be reached from the initial state, which is $(0, 0)$ here. Also notice that all terminating paths above the forbidden region are ‘equivalent’ in some sense, given that they are all characterized by the fact that T_2 gets a and b before T_1 (as far as resources are concerned, we call this a *schedule*). Similarly, all paths below the forbidden region are characterized by the fact that T_1 gets a and b before T_2 does.

On this picture, one can already recognize many ingredients that are at the center of the main problem of algebraic topology, namely the classification of shapes modulo ‘elastic deformation’. As a matter of fact, the actual coordinates that are chosen for representing the times at which P s and V s occur are unimportant, and these can be ‘stretched’ in any manner, so the properties (deadlocks, schedules etc.) are invariant under some notion of deformation, or *homotopy*. This is only a particular kind of homotopy though, and this explains why a new theory has to be designed. We call it (in subsequent work) *directed homotopy* or *dihomotopy* in the sense that it should preserve the direction of time. For example, the two homotopic shapes, all of which have two holes, of Figure 3.9 and Figure 3.10 have a different number of dihomotopy classes of dipaths. In Figure 3.9 there are essentially four dipaths up to dihomotopy (i.e., four schedules corresponding to all possibilities of accesses of resources a and b) whereas in Figure 3.10, there are essentially three dipaths up to dihomotopy [Gau02a, Gau02b, Gau02c, GG03, Gau03].

The natural combinatorial notion which discretizes this topological framework is that of a *precubical set*, which is a collection of points (states), edges (transitions), squares, cubes and hypercubes (higher-dimensional transitions representing the truly-concurrent execution of some number of actions). This was introduced in [Pra91] as well as possible formalizations using n -categories (see the next subsection), and a notion of homotopy [Lei03]. These precubical sets are called *higher-dimensional automata* (HDA) because it really makes sense to consider a hypercube as some form of transition, as in transition systems, used in semantics of programming languages.

In general, an HDA labeled over an alphabet A , is a 7-tuple (Q, d, s, t, I, F, l) where Q is a set (of states), $d : Q \rightarrow \text{Nat}$, associating to each state q a dimension $d(q)$; states of dimension 0 are called *nodes*, states of dimension 1 are *edges*, states of dimension n represent lists of n *transitions firing concurrently*, $s, t : \text{Nat}^*Q \rightarrow Q$ are partial functions ($s(k, q)$ indicates the start and $t(q, k)$

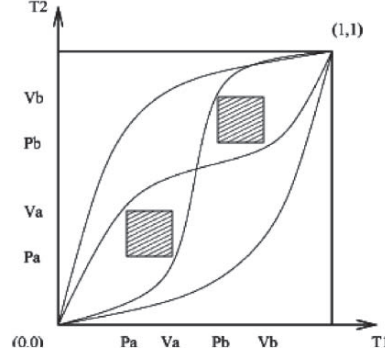


Fig. 3.9. The progress graph corresponding to $Pa.Va.Pb.Vb \mid Pa.Va.Pb.Vb$ (adapted from [Gau02a]).

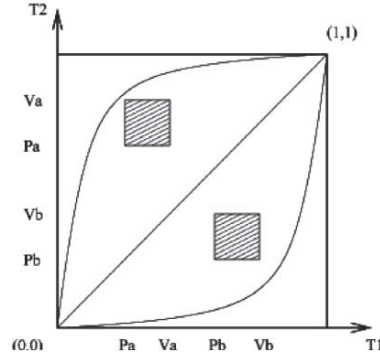


Fig. 3.10. The progress graph corresponding to $Pb.Vb.Pa.Va \mid Pa.Va.Pb.Vb$ (adapted from [Gau02a]).

the termination of the k -th transition in the list of transitions q) satisfying [Pra91]:

$$\begin{aligned}
 & s(k, q) \text{ defined iff } t(k, q) \text{ defined iff } k < d(q), \\
 & k < d(q) \geq d(s(k, q)) = d(t(k, q)) = d(q) - 1, \\
 & i \leq j \geq s(j, (s(i, q))) = s(i, s(j + 1, q)), \\
 & i \leq j \geq s(j, (t(i, q))) = t(i, s(j + 1, q)), \\
 & i \leq j \geq t(j, (s(i, q))) = s(i, t(j + 1, q)), \\
 & i \leq j \geq t(j, (t(i, q))) = t(i, t(j + 1, q)).
 \end{aligned}$$

These requirements say that $s(k,)$ and $t(k,)$ applied to a state q representing a list of $d(q) > k$ transitions yield states representing lists of $d(q) - 1$ transitions obtained by leaving out transition number k and renumbering the transitions with number $> k$. $s(0,)$ and $t(0,)$ applied on edges just yield the beginning and ending nodes of those edges. Also, I is an element of Q with $d(I) = 0$ (the

initial state), F is a subset of Q whose members q satisfy $d(q) = 0$ (the set of final states), $l : Q \rightarrow A$ is a partial function (the labeling function) defined on states of dimension 1, satisfying [Pra91]:

$$d(q) = 2 \text{ and } (k = 0 \text{ or } k = 1) \text{ imply } l(s(k, q)) = l(t(k, q)).$$

This requirement says that opposite edges in a square have the same label. This because they represent the same transition, scheduled either before or after the firing of another transition. Transitions or events can be defined as equivalence classes of edges (with respect to the finest equivalence identifying $s(k, q)$ and $t(k, q)$ for any q with $d(q) = 2$ and $(k = 0 \text{ or } k = 1)$), and represent occurrences of actions (elements of A) indicated by their label.

Let (Q, d, s, t, I, F, l) be an HDA. The *projection function* $p : \text{Nat}^*Q \rightarrow Q$ is a partial function defined by $p(k, q) = s(0, s(1, \dots s(k-1, s(k+1, \dots s(d(q)-1, q) \dots)))$ if $k < d(q)$, and $p(k, q)$ undefined if $k \geq d(q)$. Clearly, $d(p(k, q)) = 1$ and $s(0, d(p(i, q))) = s(0, d(p(j, q)))$ for $i, j, k < d(q)$.

Now the labeling function can be extended to a total function $l : Q \rightarrow A^*$ by defining:

$$\begin{aligned} l(q) &= l(p(0, q))l(s(0, q)) \text{ if } d(q) > 1, \\ \text{and } l(q) &= \text{the empty string if } d(q) = 0. \end{aligned}$$

A *path* in a higher-dimensional automaton (Q, d, s, t, I, F, l) is a sequence $p(0)p(1)\dots p(n)$ in $Q^*(n \geq 0)$ such that for any $0 < i < n+1$ there is a k in Nat such that $p(i-1) = s(k, p(i))$ or $p(i) = t(k, p(i-1))$.

Two paths P and P' are adjacent, denoted $P \leftrightarrow P'$ if there are paths Q, R and states p, p' such that $d(p)$ differs from $d(p')$ and $P = QpR$, while $P' = Qp'R$ [Pra91].

A path P' is an extension of a path P , denoted $P < P'$ if there is a path Q with $P' = PQ$.

A *bisimulation* between two higher-dimensional automata (Q, d, s, t, I, F, l) and $(Q', d', s', t', I', F', l')$ labeled over A is a binary relation R between their paths, such that their initial states (viewed as paths) are related, as well as:

- if $(p(0)p(1)\dots p(n)) R (q(0)q(1)\dots q(m))$ then $n = m$, $p(n) \in F$ iff $q(n) \in F'$, and for $0 \leq k \leq n : l(p(k)) = l'(q(k))$ (and thus $d(p(k)) = d'(q(k))$),
- if PRQ and $P \leftrightarrow P'$ then there is a Q' with $Q \leftrightarrow Q'$ and $P'RQ'$,
- if PRQ and $Q \leftrightarrow Q'$ then there is a P' with $P \leftrightarrow P'$ and $P'RQ'$,
- if PRQ and $P < P'$ then there is a Q' with $Q < Q'$ and $P'RQ'$,
- if PRQ and $Q < Q'$ then there is a P' with $P < P'$ and $P'RQ'$.

Two higher-dimensional automata are called *bisimulation equivalent* if there exists a bisimulation between them. Note that bisimulation equivalence is an equivalence indeed. For 1D automata (with $d(q) < 2$ for all $q \in Q$) a path is just an alternating sequence of nodes and edges – each edge going from the node before it to the node after it, no two paths are adjacent and the notion of bisimulation equivalence defined above coincides with the classical one [Pra91].

Let $A = (Q, d, s, t, I, F, l)$ be an HDA. *Homotopy* is the smallest equivalence relation on the paths of A containing adjacency. The unfolding $U(A)$ of A is defined as the automaton $(Q', d', s', t', I', F', l')$, where: Q' is the set of all paths in A starting at I , modulo homotopy, $d'[p(1)...p(n)] = d(p(n))$; $s'([p(1)...p(n)], k) = \{q(1)...q(n-1) | q(1)...q(n-1)p(n) \text{ is homotopic with } p(1)...p(n) \text{ and } s(p(n), k) = q(n-1)\}$; $t'([p(1)...p(n)], k) = [p(1)...p(n)t(p(n), k)]$; $I' = [I]$; $F' = \{[p(1)...p(n)] | p(n) \text{ in } F\}$; and $l'[p(1)...p(n)] = l(p(n))$. It is straightforward to check that $U(A)$ is well-defined and is an automata indeed [Pra91].

Two HDA (Q, d, s, t, I, F, l) and $(Q', d', s', t', I', F', l')$ labeled over A are *isomorphic* if there exists a bijection $f : Q \rightarrow Q'$ (an isomorphism) satisfying [Pra91]:

$$\begin{aligned} d'(f(q)) &= d(q), f(I) = I', f(q) \text{ in } F' \text{ iff } q \in F, l'(f(q)) = l(q); \\ \{s'(k, f(q)) | k < d(q)\} &= \{f(s(k, q)) | k < d(q)\}; \text{ and} \\ \{t'(k, f(q)) | k < d(q)\} &= \{f(t(k, q)) | k < d(q)\}. \end{aligned}$$

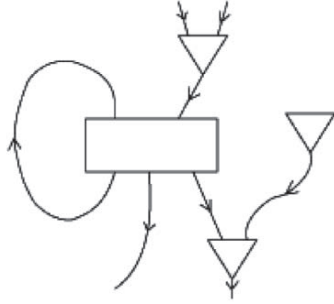
The weak requirements for s and t allow to change the order in the list of transitions represented by a state.

Working with dihomotopy classes of HDA allows to solve the *state-space explosion problem*: the number of spaces to be traversed might be exponential in the number of processes involved (see [Gau02a, Gau02b, Gau02c, GG03, Gau03]).

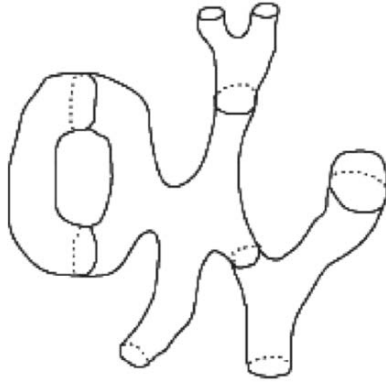
Natural Geometrical Structures

Closely related to the higher-dimensional automata are various *natural geometrical structures*, most of which are commonly called tangles.

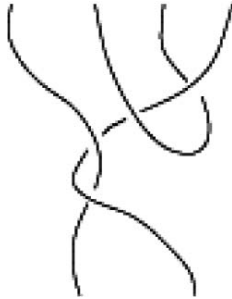
For example, a 2D flow-chart-like complex 1D-structure could be a diagram of the form [Lei02, Lei03]:



Its 3D-generalization is a surface diagram with the same information-flow:



Moreover, if we allow crossings, as in a *braid*:



then we start getting pictures that look like *knots* which are again related to higher categorical structures [Lei02, Bae97].

A category $\mathcal{C}$ with only one object is a *monoid* (= semigroup with unit) M . A 2-category $\mathcal{C}$ with only one 0-cell is a *monoidal category* $\mathcal{M}$. A *braided monoidal category* is a monoidal category equipped with a map called *braiding*

$$A \otimes B \xrightarrow{\beta_{A,B}} B \otimes A,$$

for each pair A, B of objects.

The canonical example of a braided monoidal category is $\mathcal{BR}$ [Lei03]. This has:

1. Objects: natural numbers $0, 1, \dots$;
2. Morphisms: braids, e.g.,

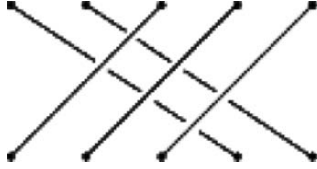


(taken up to deformation);

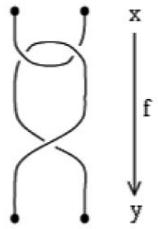
there are no morphisms $m \longrightarrow n$ when $m \neq n$;

3. Tensor product: placing side-by-side (which on objects means addition); and

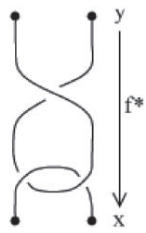
4. Braiding: right over left, e.g.,



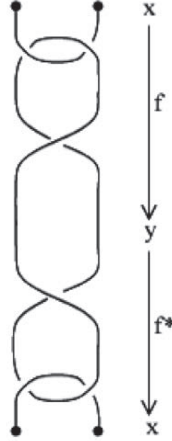
Knots, links and braids are all special cases of *tangles* (see [RT90]). The mysterious relationships between topology, algebra and physics amount in large part to the existence of interesting functors from various topologically defined categories to **Hilbert**, the category of Hilbert spaces. These topologically defined categories are always $\ast$ -categories, and the really interesting functors from them to **Hilbert** are always $\ast$ -functors, which preserve the $\ast$ -structure. Physically, the $\ast$ operation corresponds to *reversing the direction of time*. For example, there is a $\ast$ -category whose objects are collections of points and whose morphisms are tangles (see [Bae97, BD98]):



We can think of this morphism $f : x \rightarrow y$ as representing the trajectories of a collection of particles and antiparticles, where particles and antiparticles can be created or annihilated in pairs. Reversing the direction of time, we get the ‘dual’ morphism $f^* : y \rightarrow x$:



This morphism is not the inverse of f , since the composite $f \circ f^*$ is a nontrivial tangle:



Indeed, any groupoid becomes a $*$ -category if we set $f^* = f^{-1}$ for every morphism f .

The above example involves 1D curves in 3D space. More generally, topological quantum field theory studies n D manifolds embedded in $(n + k)$ D space-time, which in the $k \rightarrow \infty$ limit appear as ‘abstract’ n D manifolds. It appears that these are best described using certain ‘ n -categories with duals’, meaning n -categories in which every j -morphism f has a dual f^* .

Therefore, a tangle is a box in 3D space with knotted and linked string embedded within it and a certain number of strands of that string emanating from the surface of the box. There are no open ends of string inside the box. We usually think of some subset of the strands as *inputs* to the tangle and the remaining strands as the *outputs* from the tangle. Usually the inputs are arranged to be drawn vertically and so that they enter tangle from below, while the outputs leave the tangle from above. The tangle itself (within the box) is arranged as nicely as possible with respect to a vertical direction. This means that a definite vertical direction is chosen, and that the tangle intersects planes perpendicular to this direction transversely except for a finite collection of critical points. These basic critical points are local maxima and local minima for the space curves inside the tangle. Two tangles configured with respect to the same box are *ambient isotopic* if there is an isotopy in three space carrying one to the other that fixes the input and output strands of each tangle. We can *compose* two tangles A and B where the number of output strand of A is equal to the number of input strands of B . Composition is accomplished by joining each output strand of A to a corresponding input strand of B [KR95, KR99, Kau94].

A *tangle diagram* is a box in the plane, arranged parallel to a chosen vertical direction with a left-right ordered sequence of input strands entering the bottom of the box, and a left-right ordered sequence of output strands emanating from the top of the box. Inside the box is a diagram of the tangle represented with crossings (broken arc indicating the undercrossing line) in

the usual way for knot and links. We assume, as above, that the tangle is represented so that it is transverse to lines perpendicular to the vertical except for a finite number of points in the vertical direction along the tangle. It is said that the *tangle is well arranged, or Morse* with respect to the vertical direction when these transversality conditions are met. At the critical points we will see a local maximum, a local minimum or a crossing in the diagram. Tangle composition is well-defined (for matching input/output counts) since the input and output strands have an ordering (from left to right for the reader facing the plane on which the tangle diagram is drawn). Note that the cardinality of the set of input strands or output strands can be equal to zero. If they are both zero, then the tangle is simply a knot or link diagram arranged well with respect to the vertical direction [KR95, KR99, Kau94].

The *Reidemeister moves* are a set of moves on diagrams that combinatorially generate isotopy for knots, links and tangles [Rei48]. If two tangles are equivalent in 3D space, then corresponding diagrams of these tangles can be obtained one from another, by a sequence of Reidemeister moves. Each move is confined to the tangle box and keeps the input and output strands of the tangle diagram fixed.

Two (tangle) diagrams are said to be *regularly isotopic* if one can be obtained from the other by a sequence of Reidemeister moves of type 0,2,3 (move number 1 is not used in regular isotopy).

If A and B are given tangles, we denote the composition of A and B by AB where the diagram of A is placed below the diagram of B and the output strands of A are connected to the input strands of B . If the cardinalities of the sets of input and output strands are zero, then we simply place one tangle below the other to form the product [KR95, KR99, Kau94].

Along with tangle composition, as defined in the previous paragraph, we also have an operation of *product* or *juxtaposition* of tangles. To juxtapose two tangles A and B simply place their diagrams side by side with A to the left of B and regard this new diagram as a new tangle whose inputs are the inputs of A followed by the inputs of B , and whose outputs are the outputs of A followed by the outputs of B . We denote the tangle product of A and B by $A \otimes B$.

It remains to describe the equivalence relation on tangles that makes them represent regular isotopy classes of embedded string. Every tangle is a composition of *elementary tangles* where an elementary tangle is one of the following list: a *cup* (a single minimum – zero inputs, two outputs), a *cap* (a single maximum – two inputs, zero outputs), a *crossing* (a single local crossing diagram – two inputs and two outputs).

Ultimate Conceptual Machines: Weak n –Categories

As traditionally conceived, an n –category is an algebraic structure having objects or 0–morphisms, 1–morphisms between 0–morphisms, 2–morphisms between 1–morphisms, and so on up to n –morphisms. There should be

various ways of composing j -morphisms, and these composition operations should satisfy various laws, such as associativity laws. In the so-called *strict n -categories*, these laws are equations. While well-understood and tractable, strict n -categories are insufficiently general for many applications: what one usually encounters in nature are *weak n -categories*, in which composition operations satisfy the appropriate laws *only up to equivalence*. Here the idea is that n -morphisms are equivalent precisely when they are equal, while for $j < n$ an equivalence between j -morphisms is recursively defined as a $(j + 1)$ -morphism from one to the other that is invertible up to equivalence [BD98].

Now, what makes it difficult to define weak n -categories is that laws formulated as equivalences should satisfy laws of their own – the so-called *coherence laws* – so that one can manipulate them with some of the same facility as equations. Moreover, these coherence laws should also be equivalences satisfying their own coherence laws, again up to equivalence, and so on [BD98].

For example, a weak 1-category is just an ordinary category. In a category, composition of 1-morphisms is associative:

$$(fg)h = f(gh).$$

Weak 2-categories first appeared in the work of Bénabou [Ben67], under the name of *bicategories*. In a bicategory, composition of 1-morphisms is associative only up to an invertible 2-morphism, the ‘*associator*’:

$$A_{f,g,h}: (fg)h \rightarrow f(gh).$$

The associator allows one to re-bracket parenthesized composites of arbitrarily many 1-morphisms, but there may be many ways to use it to go from one parenthesization to another. For all these to be equal, the associator must satisfy a coherence law, the *pentagon identity*, which says that the following diagram commutes:

$$\begin{array}{ccc} ((fg)h)i & \xrightarrow{(fg)(hi)} & f(g(hi)) \\ \downarrow & & \uparrow \\ (f(gh))i & \xrightarrow{\quad\quad\quad} & f((gh)i) \end{array}$$

where all the arrows are 2-morphisms built using the associator. Weak 3-categories or *tricategories* were defined by [GPS95]. In a tricategory, the pentagon identity holds only up to an invertible 3-morphism, which satisfies a further coherence law of its own.

When one explicitly lists the coherence laws this way, the definition of weak n -category tends to grow ever more complicated with increasing n . To get around this, one must carefully study the origin of these coherence laws.

So far, most of our insight into coherence laws has been won through homotopy theory, where it is common to impose equations *only up to homotopy*, with these homotopies satisfying coherence laws, again up to homotopy, and so on. For example, the pentagon identity and higher coherence laws for associativity first appeared in Stasheff's work on the structure inherited by a space equipped with a homotopy equivalence to a space with an associative product [Sta63]. Subsequent work led to a systematic treatment of coherence laws in homotopy theory through the formalism of *topological operads* [Ada78].

Underlying the connection between homotopy theory and n -category theory is a hypothesis made quite explicit by Grothendieck [Gro83]: to any topological space one should be able to associate an n -category having points as objects, paths between points as 1-morphisms, certain paths of paths as 2-morphisms, and so on, with certain homotopy classes of n -fold paths as n -morphisms. This should be a special sort of weak n -category called a *weak n -groupoid*, in which all j -morphisms ($0 < j \leq n$) are equivalences. Moreover, the process of assigning to each space its *fundamental n -groupoid*, as Grothendieck called it, should set up a complete correspondence between the theory of homotopy n -types (spaces whose homotopy groups vanish above the n th) and the theory of weak n -groupoids. This hypothesis explains why all the coherence laws for weak n -groupoids should be deducible from homotopy theory. It also suggests that weak n -categories will have features not found in homotopy theory, owing to the presence of j -morphisms that are not equivalences [BD98].

Homotopy theory also makes it clear that when setting up a theory of n -categories, there is some choice involved in the shapes of ones j -morphisms – or in the language of topology, j -cells. The traditional approach to n -categories is *globular*. This means that for $j > 0$, each j -cell $f: x \rightarrow y$ has two $(j-1)$ -cells called its *source*, $sf = x$, and *target*, $tf = y$, which for $j > 1$ satisfy

$$s(sf) = s(tf), \quad t(sf) = t(tf).$$

Thus a j -cell can be visualized as a *globe*, a j D ball whose boundary is divided into two $(j-1)$ D hemispheres corresponding to its source and target. However, in homotopy theory, the simplicial approach is much more popular. In a *simplicial set*, each j -cell f is shaped like a j D simplex, and has $j+1$ faces, certain $(j-1)$ -cells $d_0f, \dots, d_nf$. In addition to these there are $(j+1)$ -cells $i_0f, \dots, i_{n+1}f$ called *degeneracies*, and the face and degeneracy maps satisfy certain well-known relations [BD98].

3.5.7 Brain–Mind Functorial Machines

In this subsection we present two models of the *brain–mind functorial machines*: the first one is a psychologically-motivated top-down machine, while the second one is physically-motivated bottom-up solitary machine.

Neurodynamical 2-Functor

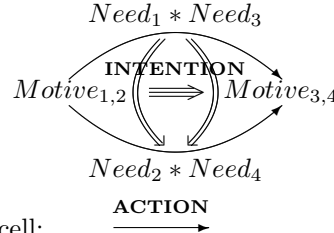
Here we define the goal-directed cognitive neurodynamics as an evolution 2-functor $\mathcal{E}$ given by

$$\begin{array}{ccc}
 \begin{array}{ccc}
 A & \xrightarrow{f} & B \\
 \downarrow h & \text{CURRENT} & \downarrow g \\
 & \text{NEURAL} & \\
 & \text{STATE} & \\
 C & \xrightarrow{k} & D
 \end{array} & \xrightarrow{\mathcal{E}} & \begin{array}{ccc}
 \mathcal{E}(A) & \xrightarrow{\mathcal{E}(f)} & \mathcal{E}(B) \\
 \downarrow \mathcal{E}(h) & \text{DESIRED} & \downarrow \mathcal{E}(g) \\
 & \text{NEURAL} & \\
 & \text{STATE} & \\
 \mathcal{E}(C) & \xrightarrow{\mathcal{E}(k)} & \mathcal{E}(D)
 \end{array}
 \end{array} \quad (3.488)$$

In (3.488), $\mathcal{E}$ represents a projection map from the *source 2-category* of the current neural state, defined as a commutative square of small categories $A, B, C, D, \dots$ of *current neural ensembles* and their causal interrelations $f, g, h, k, \dots$, onto the *target 2-category* of the desired neural state, defined as a commutative square of small categories $\mathcal{E}(A), \mathcal{E}(B), \mathcal{E}(C), \mathcal{E}(D), \dots$ of *evolved neural ensembles* and their causal interrelations $\mathcal{E}(f), \mathcal{E}(g), \mathcal{E}(h), \mathcal{E}(k)$.

The evolution 2-functor $\mathcal{E}$ can be horizontally decomposed in the following three neurodynamic components (see [Lew97, AL91]):

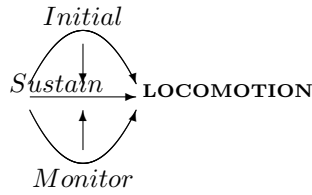
1. *Intention*, defined as a 3-cell:



2. *Action*, defined as a 1-cell:

$\xrightarrow{\text{ACTION}}$

3. *Locomotion*, defined as a 2-cell:



Now, each causal arrow in (3.488), say $f : A \rightarrow B$, stands for a generic ‘neuro-morphism’, representing a self-organized, oscillatory neurodynamic system. We define a generic neuro-morphism f to be a nonlinear tensor-field (x, y, ω) -system (3.489–3.494), acting as a bidirectional associative memory machine on a ND Riemannian manifold M^N of the human cortex. It is formed out of two distinct, yet nonlinearly-coupled neural subsystems:

1. Activation (x, y) -dynamics (3.489–3.490), defined as an interplay of an excitatory vector-field $x^i = x^i(t) : M^N \rightarrow TM$, representing a cross-section of the tangent bundle TM , and an inhibitory 1-form $y_i = y_i(t) : M^N \rightarrow T^*M$, representing a cross-section of the cotangent bundle T^*M .
2. Excitatory and inhibitory unsupervised learning (ω) -dynamics (3.491–3.494) generated by random differential Hebbian learning process (3.493–3.494), defined respectively by contravariant synaptic tensor-field $\omega^{ij} = \omega^{ij}(t) : M^N \rightarrow TTM_{im}^N$ and covariant synaptic tensor-field $\omega_{ij} = \omega_{ij}(t) : M^N \rightarrow T^*T^*M$, representing cross-sections of contravariant and covariant tensor bundles, respectively.

(x, y, ω) -system is analytically defined as a set of N coupled neurodynamic equations:

$$\dot{x}^i = A^i + \omega^{ij} f_j(y) - x^i, \quad (3.489)$$

$$\dot{y}_i = B_i + \omega_{ij} f^j(x) - y_i, \quad (3.490)$$

$$\dot{\omega}^{ij} = -\omega^{ij} + I^{ij}(x, y), \quad (3.491)$$

$$\dot{\omega}_{ij} = -\omega_{ij} + I_{ij}(x, y), \quad (3.492)$$

$$I^{ij} = f^i(x) f^j(y) + \dot{f}^i(x) \dot{f}^j(y) + \sigma^{ij}, \quad (3.493)$$

$$I_{ij} = f_i(x) f_j(y) + \dot{f}_i(x) \dot{f}_j(y) + \sigma_{ij}, \quad (3.494)$$

$$(i, j = 1, \dots, N).$$

Here ω is a symmetric, second-order synaptic tensor-field; $I^{ij} = I^{ij}(x, y, \sigma)$ and $I_{ij} = I_{ij}(x, y, \sigma)$ respectively denote contravariant-excitatory and covariant-inhibitory random differential Hebbian innovation-functions with tensorial Gaussian noise σ (in both variances); f s and $\dot{f}$ s denote sigmoid activation functions ($f = \tanh(\cdot)$) and corresponding signal velocities ($\dot{f} = 1 - f^2$), respectively in both variances; $A^i = A^i(t)$ and $B_i = B_i(t)$ are contravariant-excitatory and covariant-inhibitory neural inputs to the corresponding cortical cells, respectively;

Nonlinear activation (x, y) -dynamics, describes a two-phase biological neural oscillator field, in which the excitatory neural field excites the inhibitory neural field, which itself reciprocally inhibits the excitatory one. (x, y) -dynamics represents a nonlinear extension of a linear, Lyapunov-stable, conservative, gradient system, defined in local neural coordinates $x^i, y_i \in V_y$ on T^*M as

$$\dot{x}^i = -\frac{\partial \Phi}{\partial y_i} = \omega^{ij} y_j - x^i, \quad \dot{y}_i = -\frac{\partial \Phi}{\partial x^i} = \omega_{ij} x^j - y_i. \quad (3.495)$$

The gradient system (3.495) is derived from scalar, neuro-synaptic action potential $\Phi : T^*M \rightarrow \mathbb{R}$, given by a negative, smooth bilinear form in $x^i, y_i \in V_y$ on T^*M as

$$-2\Phi = \omega_{ij} x^i x^j + \omega^{ij} y_i y_j - 2x^i y_i, \quad (i, j = 1, \dots, N),$$

which itself represents a Ψ -image of the Riemannian metrics $g : TM \rightarrow \mathbb{R}$ on the configuration manifold M^N .

The nonlinear oscillatory activation (x, y) -dynamics (3.489–3.492) is get from the linear conservative dynamics (3.495), by adding configuration dependent inputs A^i and B_i , as well as sigmoid activation functions f_j and f^j , respectively. It represents an interconnected pair of excitatory and inhibitory neural fields.

Both variant-forms of learning (ω) -dynamics (3.491–3.492) are given by a generalized unsupervised (self-organizing) Hebbian learning scheme (see [Kos92]) in which $\dot{\omega}_{ij}$ (resp. $\dot{\omega}^{ij}$) denotes the new-update value, $-\omega_{ij}$ (resp. ω^{ij}) corresponds to the old value and $I_{ij}(x^i, y_j)$ (resp. $I^{ij}(x^i, y_j)$) is the innovation function of the symmetric 2nd order synaptic tensor-field ω . The nonlinear innovation functions I_{ij} and I^{ij} are defined by random differential Hebbian learning process (3.493–3.494). As ω is a symmetric and zero-trace coupling synaptic tensor, the conservative linear activation dynamics (3.495) is equivalent to the rule that ‘the state of each neuron (in both neural fields) is changed in time if, and only if, the scalar action potential Φ (52), is lowered’. Therefore, the scalar action potential Φ represents the monotonically decreasing Lyapunov function (such that $\dot{\Phi} \leq 0$) for the conservative linear dynamics (3.495), which converges to a local minimum or ground state of Φ . That is to say, the system (3.495) moves in the direction of decreasing the scalar action potential Φ , and when both $\dot{x}^i = 0$ and $\dot{y}_i = 0$ for all $i = 1, \dots, N$, the steady state is reached.

Solitary ‘Thought Nets’ and ‘Emerging Mind’

Synergetic ‘Thought Solitons’

Recall that synergetics teaches us that *order parameters* (and their spatio-temporal evolution) are *patterns, emerging from chaos*. In our opinion, the most important of these order parameters, both natural and man made, are *solitons*, because of their self-organizing quality to create order out of chaos. From this perspective, *nonlinearity* – the essential characteristic of nature – is the *cause* of both *chaos* and *order*. Recall that the solitary particle-waves, also called the ‘light bullets’, are localized space-time excitations $\Psi(x, t)$, propagating through a certain medium Ω with constant velocities v_j . They describe a variety of nonlinear wave phenomena in one dimension and playing important roles in optical fibers, many branches of physics, chemistry and biology.

To derive our solitary network we start with modeling the conservative ‘*thought solitons*’, using the following three classical nonlinear equations, defining the time evolution of the spatio-temporal wave function $\Psi(x, t)$ (which is smooth, and either complex-, or real-valued) (see [NMP84, For90, AC91, IP01a]):

1. Nonlinear Schrödinger (NS) equation

$$i\Psi_t = 2\mu|\Psi|^2\Psi - \Psi_{xx}, \quad (3.496)$$

where $\Psi = \Psi(x, t)$ is a complex-valued wave function with initial condition $\Psi(x, t)|_{t=0} = \Psi(x)$ and μ is a nonlinear parameter representing field strength. In the linear limit ($a = 0$) NS becomes the ordinary Schrödinger equation for the wave function of the free 1D particle with mass $m = 1/2$. Its Hamiltonian function

$$H_{NS} = \int_{-\infty}^{+\infty} \left(\mu|\Psi|^4 + |\Psi_x|^2 \right) dx,$$

is equal to the total and conserved energy of the soliton. NS describes, for example, nonlinear Faraday resonance in a vertically oscillating water, an easy-plane ferromagnet with a combination of a stationary and a high-frequency magnetic fields, and the effect of phase-sensitive amplifiers on solitons propagating in optical fibers.

2. *Korteweg-de Vries (KdV) equation*

$$\Psi_t = 6\Psi\Psi_x - \Psi_{xxx},$$

with Hamiltonian (total conserved energy) given by

$$H_{KdV} = \int_{-\infty}^{+\infty} \left(\Psi^3 + \frac{1}{2}\Psi_x^2 \right) dx.$$

KdV is related to the ordinary Schrödinger equation by the inverse scattering method. KdV is a well-known model of 1D turbulence that was derived in various physical contexts, including chemical-reaction waves, propagation of combustion fronts in gases, surface waves in a film of a viscous liquid flowing along an inclined plane, patterns in thermal convection, rapid solidification, and others. Its discretization gives the Lotka-Volterra equation

$$\dot{x}^j(t) = x^j(t) (x^{j+1}(t) - x^{j-1}(t)),$$

which appears in a model of struggle for existence of biological species.

3. *Sine-Gordon (SG) equation*

$$\Psi_{tt} = \Psi_{xx} - \sin \Psi,$$

with Hamiltonian (total conserved energy) given by

$$H_{SG} = \int_{-\infty}^{+\infty} (\Psi_t^2 + \Psi_x^2 + \cos \Psi) dx.$$

SG gives one of the simplest models of the unified field theory, can be found in the theory of dislocations in metals, in the theory of Josephson junctions and so on. It can be used also in interpreting certain biological processes like DNA dynamics. Its discretization gives a *system of coupled pendulums*.

Discrete solitons exist also in the form of the *soliton cellular automata* (SCA) [PST86]. SCA is a 1(space)+1(time)-dimensional ‘box and ball system’ made of infinite number of zeros (or, boxes) and finite number of ones (or, balls). The value of the j th SCA cell a_t^j at a discrete time t , is given as

$$a_{t+1}^j = \begin{cases} 1, & \text{if } a_t^j = 0 \text{ and } \sum_{i=-\infty}^{j-1} u_t^i > \sum_{i=-\infty}^{j-1} a_{t+1}^i, \\ 0, & \text{otherwise,} \end{cases}$$

where $a_t^j = 0$ is assumed for $|j| \gg 1$. Any state of the SCA consists purely of solitons (particularly, KdV-solitons), possessing conserved quantities of the form of H_{KdV} . All of these properties have motivated a number of suggestive applications for a new kind of computational architecture that will use these evolution patterns of SCA in order to give a ‘gateless’ implementation of logical operations.

In practice, both SCA and KdV are usually approximated by the *Toda lattice equation*,

$$\ddot{q}^i = e^{q^{i+1}-q^i} - e^{q^i-q^{i-1}}, \quad (i = 1, \dots, N) \quad (3.497)$$

with quasiperiodic $q^{N+i}(t) = q^i(t) + c$, or,
fast-decaying boundary conditions $\lim_{i \rightarrow -\infty} q^i(t) = 0$, $\lim_{i \rightarrow +\infty} q^i(t) = c$.

The Toda equation (3.497) is a *gradient Newtonian equation* of motion

$$\ddot{q}^i = -\partial_q^i V, \quad V(q) = \sum_{i=1}^N e^{q^{i+1}-q^i}.$$

Otherwise, the Toda equation represents a Hamiltonian system

$$\dot{q}^i = p_i, \quad \dot{p}_i = e^{q^{i+1}-q^i} - e^{q^i-q^{i-1}},$$

with the phase-space $P = \mathbb{R}^{2N}$ with coordinates (p_i, q^i) , standard Poisson structure

$$\{p_i, p_j\} = \{q^i, q^j\} = 0, \quad \{p_i, q^j\} = \delta_i^j,$$

and Hamiltonian function $H = \sum_{i=1}^N \left(\frac{1}{2} p_i^2 + e^{q^{i+1}-q^i} \right)$, $(i, j = 1, \dots, N)$.

Next, to make our conservative thought solitons *open to the environment*, we have to modify them by adding:

1. Input from the senses, in the form of the Weber-Fechner’s law,

$$S(t) = a_r \log s^r(t), \quad (r = 1, \dots, 5), \quad (3.498)$$

where $S = S(t)$ is the *sensation*, $s^r = s^r(t)$ the vector of stimuli from the five senses, and a_r a constant vector; and

2. Disturbances, in the form of additive, zero-mean Gaussian white noise $\eta = \eta(t)$, independent from the main soliton-signal.

In this way, we get the *modified solitary equations*:

$$\text{MNS : } i\Psi_t = 2\mu|\Psi|^2\Psi - \Psi_{xx} + a_r \log s^r \Psi + \eta,$$

$$\text{MKdV : } \Psi_t = 6\Psi\Psi_x - \Psi_{xxx} + a_r \log s^r \Psi + \eta,$$

$$\text{MSG : } \Psi_{tt} = \Psi_{xx} - \sin \Psi + a_r \log s^r \Psi + \eta,$$

representing the *three different models of the thought units*.

Now we will form a single emerging order-parameter, the general factor, that we call the *Mind*. It behaves like an orchestrated ensemble of thought solitons, defined as systems of trainable, coupled soliton equations. Their tensor couplings perform *self-organizing associative learning by trial and error*, similar to that of the neural ensemble.

The dynamics of the *soliton ensemble*, representing our model of the ‘mind’ can be described as one of the following three soliton systems; each of them performs *learning*, *growing* and *competing* between each other, and *communicates with environment* through the *sensory inputs* and the *heating noise*:

1. Coupled modified nonlinear Schrödinger equations

$$\begin{aligned} i\Psi_t^k &= -\Psi_{xx}^k + 2\mu^k \sum_{j \neq k} |\Psi^j|^2 \mathbf{W}_k^j S^j(\Psi^j) \\ &+ \nu^k \Psi^k (1 - \epsilon^k \Psi^k) + a_r \log s^r \Psi^k + \eta^k, \end{aligned}$$

2. Coupled modified Korteweg-de Vries equations

$$\begin{aligned} \Psi_t^k &= 6\Psi_x^k \Psi^k - \Psi_{xxx}^k + \sum_{j \neq k} \mathbf{W}_k^j S^j(\Psi^j) \\ &+ \nu^k \Psi^k (1 - \epsilon^k \Psi^k) + a_r \log s^r \Psi^k + \eta^k, \end{aligned}$$

3. Coupled modified Sine-Gordon equations

$$\Psi_{tt}^k = \Psi_{xx}^k - \sin \Psi^k + \sum_{j \neq k} \nu^k \Psi^k (1 - \epsilon^k \Psi^k) + a_r \log s^r \Psi^k + \eta^k,$$

where $\Psi^k = \Psi^k(x, t)$, ($k = 1, \dots, n$) is the set of wave functions of the solitary thoughts, $S(\cdot)$ represents the sigmoidal threshold functions, ν^k and ϵ^k are *growing* and *competition* parameters.

$\mathbf{W}_k^j = \mathbf{W}_k^j(\Psi)$ are *tensorial learning couplings*, evolving according to the Hebbian learning scheme (see [Kos92]):

$$\dot{\mathbf{W}}_k^j = -\mathbf{W}_k^j + \Phi_k^j(\Psi^k, \Psi^j),$$

with *innovation* defined in tensor signal form (here $\dot{S}(\cdot) = 1 - \tanh(\cdot)$)

$$\Phi_k^j = S^j(\Psi^j) S^k(\Psi^k) + \dot{S}^j(\Psi^j) \dot{S}^k(\Psi^k).$$

3.5.8 Path-Integral Psychodynamics

In this section, which is written in the fashion of the *quantum brain*, we present the top level of natural biodynamics, using geometrical generalization of the *Feynman path integral*. To formulate the basics of *force-field psychodynamics*, we use the *action-amplitude picture* of the $BODY \rightleftharpoons MIND$ adjunction:

↓ **Deterministic (causal) world of *Human BODY*** ↓

$$Action : S[q^n] = \int_{t_{in}}^{t_{out}} (E_k - E_p + Wrk + Src^\pm) dt$$

$$Amplitude : \langle out|in \rangle = \oint \mathcal{D}[w_n q^n] e^{iS[q^n]}$$

↑ **Probabilistic (fuzzy) world of *Human MIND*** ↑

In the action integral, E_k, E_p, Wrk and $Src^\pm$ denote the kinetic and potential energies, work done by dissipative/driving forces and other energy sources/sinks, respectively. In the amplitude integral, the peculiar sign $\oint$ denotes integration along smooth paths and summation along discrete Markov chains; i is the imaginary unit, w_n are synaptic-like weights, while $\mathcal{D}$ is the Feynman path differential (defined below) calculated along the configuration trajectories q^n . The action $S[q^n]$, through the *least action principle* $\delta S = 0$, leads to all biodynamic equations considered so far (in generalized Lagrangian and Hamiltonian form). At the same time, the action $S[q^n]$ figures in the exponent of the path integral $\oint$, defining the probability transition amplitude $\langle out|in \rangle$. In this way, the whole body dynamics is incorporated in the mind dynamics. This *adaptive path integral* represents an *infinite-dimensional neural network*, suggesting an infinite capacity of human brain/mind.

For a long time the cortical systems for *language and actions* were believed to be independent modules. However, according to the recent research of [Pul05], as these systems are reciprocally connected with each other, information about language and actions might interact in distributed neuronal assemblies. A critical case is that of action words that are semantically related to different parts of the body (e.g. ‘pick’, ‘kick’, ‘lick’,...). The author suggests that the comprehension of these words might specifically, rapidly and automatically activate the motor system in a somatotopic manner, and that their comprehension rely on activity in the action system.

Motivational Cognition in the Life Space Foam

Applications of nonlinear dynamical systems (NDS) theory in psychology have been encouraging, if not universally productive/effective [Met97]. Its historical

antecedents can be traced back to Piaget's [PHE92] and Vygotsky's [Vyg82] interpretations of the dynamic relations between action and thought, Lewinian theory of social dynamics and cognitive–affective development [Lew51, Gol99], and Bernstein's [Ber47] theory of self-adjusting, goal-driven motor action.

Now, both the original *Lewinian force-field theory* in psychology (see [Lew51, Gol99]) and modern decision-field dynamics (see [BT93, RBT01, BD02]) are based on the classical Lewinian concept of an individual's *life space*.¹⁵ As a topological construct, Lewinian life space represents a person's psychological environment that contains *regions* separated by dynamical permeable *boundaries*. As a field construct, on the other hand, the life space is not empty: each of its regions is characterized by *valence* (ranging from positive or negative and resulting from an interaction between the person's *needs* and the dynamics of their *environment*). Need is an energy construct, according to Lewin. It creates *tension* in the person, which, in combination with other tensions, initiates and sustains behavior. Needs vary from the most primitive urges to the most idiosyncratic intentions and can be both internally generated (e.g., thirst or hunger) and stimulus-induced (e.g., an urge to buy something in response to a TV advertisement). Valences are, in essence, personal values dynamically derived from the person's needs and attached to various regions in their life space. As a field, the life space generates forces pulling the person towards positively-valenced regions and pushing them away from regions with negative valence. Lewin's term for these forces is *vectors*. Combinations of multiple vectors in the life space cause the person to move from one region towards another. This movement is termed *locomotion* and it may range from overt behavior to cognitive shifts (e.g., between alternatives in a decision-making process). Locomotion normally results in crossing the boundaries between regions. When their permeability is degraded, these boundaries become *barriers* that restrain locomotion. Life space model, thus, offers a meta-theoretical language to describe a wide range of behaviors, from goal-directed action to intrapersonal conflicts and multi-alternative decision-making.

In order to formalize the Lewinian life-space concept, a set of *action principles* need to be associated to Lewinian force-fields, (loco)motion paths (representing mental abstractions of biomechanical paths [Iva04]) and life space geometry. As an extension of the Lewinian concept, in this paper we introduce a new concept of *life-space foam* (LSF, see Figure 3.11). According to this new concept, Lewin's life space can be represented as a *geometrical functor* with globally smooth macro-dynamics, which is at the same time underpinned by wildly fluctuating, non-smooth, local micro-dynamics, describable by *Feynman's*: (i) *sum-over-histories* $\mathcal{F}_{paths}$, (ii) *sum-over-fields* $\mathcal{F}_{fields}$, and (iii) *sum-over-geometries* $\mathcal{F}_{geom}$.

¹⁵ The work presented in this subsection has been developed in collaboration with Dr. Eugene Aidman, Senior Research Scientist, Human Systems Integration, Land Operations Division, Defence Science & Technology Organisation, Australia.

LSF is thus a two-level *geometro-dynamical functor*, representing these two distinct types of dynamics within the Lewinian life space. At its *macroscopic spatio-temporal level*, LSF appears as a ‘nice and smooth’ geometrical functor with globally predictable dynamics – formally, a smooth n -dimensional manifold M with local Riemannian metrics $g_{ij}(x)$, smooth force-fields and smooth (loco)motion paths, as conceptualized in the Lewinian theory. To model the global and smooth macro-level LSF-paths, fields and geometry, we use the general physics-like *principle of the least action*.

Now, the apparent smoothness of the macro-level LSF is achieved by the existence of another level underneath it. This *micro-level* LSF is actually a collection of wildly fluctuating force-fields, (loco)motion paths, curved regional geometries and topologies with holes. The micro-level LSF is proposed as an extension of the Lewinian concept: it is characterized by uncertainties and fluctuations, enabled by microscopic time-level, microscopic transition paths, microscopic force-fields, local geometries and varying topologies with holes. To model these fluctuating microscopic LSF-structures, we use three instances of *adaptive path integral*, defining a multi-phase and multi-path (also multi-field and multi-geometry) *transition* process from *intention* to the goal-driven *action*.

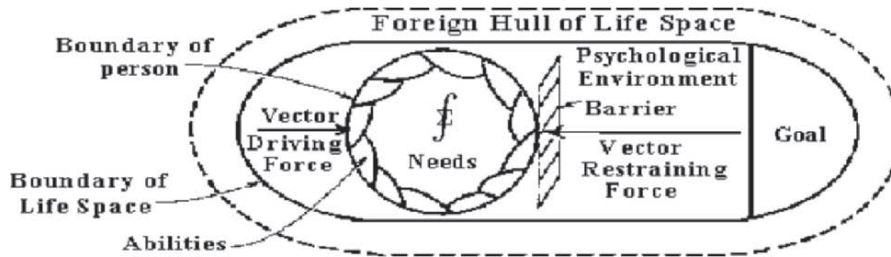


Fig. 3.11. Diagram of the *life space foam*: Lewinian life space with an adaptive path integral acting inside it and generating microscopic fluctuation dynamics.

We use the new LSF concept to develop modeling framework for motivational dynamics (MD) and induced cognitive dynamics (CD).

According to Heckhausen (see [Hec77]), *motivation* can be thought of as a process of *energizing* and *directing the action*. The process of energizing can be represented by Lewin’s *force-field analysis* and Vygotsky’s *motive formation* (see [Vyg82, AL91]), while the process of directing can be represented by *hierarchical action control* (see [Ber47, Ber35, Kuh85]).

Motivation processes both precede and coincide with every goal-directed action. Usually these motivation processes include the sequence of the following four feedforward *phases* [Vyg82, AL91]: (*)

1. *Intention Formation $\mathcal{F}$* , including: decision making, commitment building, etc.

2. *Action Initiation* $\mathcal{I}$, including: handling conflict of motives, resistance to alternatives, etc.
3. *Maintaining the Action* $\mathcal{M}$, including: resistance to fatigue, distractions, etc.
4. *Termination* $\mathcal{T}$, including parking and avoiding addiction, i.e., staying in control.

With each of the phases $\{\mathcal{F}, \mathcal{I}, \mathcal{M}, \mathcal{T}\}$ in (*), we can associate a *transition propagator* – an ensemble of (possibly crossing) feedforward paths propagating through the ‘wood of obstacles’ (including topological holes in the LSF, see Figure 3.12), so that the complete *transition functor* $\mathcal{TA}$ is a product of propagators (as well as sum over paths). All the phases–propagators are controlled by a unique *Monitor* feedback process.

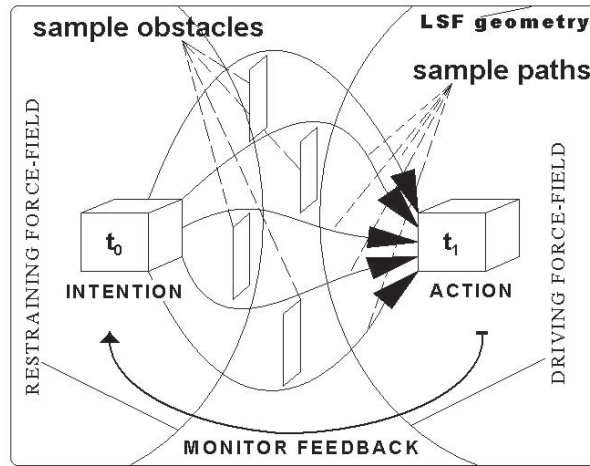


Fig. 3.12. *Transition-propagator* corresponding to each of the motivational phases $\{\mathcal{F}, \mathcal{I}, \mathcal{M}, \mathcal{T}\}$, consisting of an ensemble of feedforward paths propagating through the ‘wood of obstacles’. The paths affected by driving and restraining force-fields, as well as by the local LSF-geometry. Transition goes from *Intention*, occurring at a sample time instant t_0 , to *Action*, occurring at some later time t_1 . Each propagator is controlled by its own *Monitor* feedback. All together they form the transition functor $\mathcal{TA}$.

In this subsection we propose an *adaptive path integral* formulation for the motivational–transition functor $\mathcal{TA}$. In essence, we sum/integrate over different paths and make a product (composition) of different phases–propagators. Recall that this is the most general description of the general *Markov stochastic process*.

We will also attempt to demonstrate the utility of the same LSF-formalisms in representing cognitive functions, such as memory, learning and decision

making. For example, in the classical *Stimulus encoding* $\rightarrow$ *Search* $\rightarrow$ *Decision* $\rightarrow$ *Response* sequence [Ste69, Ash94], the environmental input-triggered *sensory memory* and *working memory* (WM) can be interpreted as operating at the micro-level force-field under the executive control of the *Monitor* feedback, whereas *search* can be formalized as a *control* mechanism guiding retrieval from the long-term memory (LTM, itself shaped by learning) and filtering material relevant to decision making into the WM. The essential measure of these mental processes, the *processing speed* (essentially determined by Sternberg's reaction-time) can be represented by our (loco)motion speed $\dot{x}$.

Six Faces of the Life Space Foam

The LSF has three forms of appearance: *paths* + *field* + *geometries*, acting on both macro-level and micro-level, which is six modes in total. In this section, we develop three least action principles for the macro-LSF-level and three adaptive path integrals for the micro-LSF-level. While developing our psycho-physical formalism, we will address the behavioral issues of motivational fatigue, learning, memory and decision making.

General Formalism

At both macro- and micro-levels, the total LSF represents a union of transition paths, force-fields and geometries, formally written as

$$\begin{aligned} LSF_{total} &:= LSF_{paths} \bigcup LSF_{fields} \bigcup LSF_{geom} \\ &\equiv \mathcal{F}_{paths} + \mathcal{F}_{fields} + \mathcal{F}_{geom}. \end{aligned} \quad (3.499)$$

Corresponding to each of the three LSF-subspaces in (3.499) we formulate:

1. The *least action principle*, to model deterministic and predictive, macro-level MD and CD, giving a unique, global, causal and smooth path-field-geometry on the macroscopic spatio-temporal level; and
2. Associated *adaptive path integral* to model uncertain, fluctuating and probabilistic, micro-level MD and CD, as an ensemble of local paths-fields-geometries on the microscopic spatio-temporal level, to which the global macro-level MD and CD represents both time and ensemble *average* (which are equal according to the *ergodic hypothesis*).

In the proposed formalism, transition paths $x^i(t)$ are affected by the force-fields $\varphi^k(t)$, which are themselves affected by geometry with metric g_{ij} .

Global Macro-Level of LSF_{total} . In general, at the *macroscopic* LSF-level we first formulate the *total action* $S[\Phi]$, the central quantity in our formalism that has psycho-physical dimensions of *Energy* $\times$ *Time* = *Effort*, with immediate cognitive and motivational applications: *the greater the action – the higher the speed of cognitive processes and the lower the macroscopic fatigue* (which includes all sources of physical, cognitive and emotional fatigue

that influence motivational dynamics). The action $S[\Phi]$ depends on macroscopic paths, fields and geometries, commonly denoted by an abstract field symbol Φ^i . The action $S[\Phi]$ is formally defined as a temporal integral from the *initial* time instant t_{ini} to the *final* time instant t_{fin} ,

$$S[\Phi] = \int_{t_{ini}}^{t_{fin}} \mathcal{L}[\Phi] dt, \quad (3.500)$$

with *Lagrangian density* given by

$$\mathcal{L}[\Phi] = \int d^n x \mathcal{L}(\Phi_i, \partial_{x^j} \Phi^i),$$

where the integral is taken over all n coordinates $x^j = x^j(t)$ of the LSF, and $\partial_{x^j} \Phi^i$ are time and space partial derivatives of the Φ^i -variables over coordinates.

Second, we formulate the *least action principle* as a minimal variation δ of the action $S[\Phi]$

$$\delta S[\Phi] = 0, \quad (3.501)$$

which, using techniques from the calculus of variations gives, in the form of the so-called Euler–Lagrangian equations, a shortest (loco)motion path, an extreme force-field, and a life-space geometry of minimal curvature (and without holes). In this way, we effectively derive a *unique globally smooth transition functor*

$$\mathcal{TA} : INTENTION_{t_{ini}} \Rightarrow ACTION_{t_{fin}}, \quad (3.502)$$

performed at a macroscopic (global) time-level from some initial time t_{ini} to the final time t_{fin} .

In this way, we get macro-objects in the global LSF: a single path described Newtonian-like equation of motion, a single force-field described by Maxwellian-like field equations, and a single obstacle-free Riemannian geometry (with global topology without holes).

For example, recall that in the period 1945–1949 J. Wheeler and R. Feynman developed their *action-at-a-distance electrodynamics* [WF49], in complete experimental agreement with the classical Maxwell’s electromagnetic theory, but at the same time avoiding the complications of divergent self-interaction of the Maxwell’s theory as well as eliminating its infinite number of field degrees of freedom. In Wheeler–Feynman view, “Matter consists of electrically charged particles,” so they found a form for the action directly involving the motions of the charges only, which upon variation would give the Newtonian-like equations of motion of these charges. Here is the expression for this action in the flat space-time, which is in the core of quantum electrodynamics:

$$S[x; t_i, t_j] = \frac{1}{2} m_i \int (\dot{x}_\mu^i)^2 dt_i + \frac{1}{2} e_i e_j \int \int \delta(I_{ij}^2) \dot{x}_\mu^i(t_i) \dot{x}_\mu^j(t_j) dt_i dt_j$$

with

$$I_{ij}^2 = [x_\mu^i(t_i) - x_\mu^j(t_j)] [x_\mu^i(t_i) - x_\mu^j(t_j)], \quad (3.503)$$

where $x_\mu^i = x_\mu^i(t_i)$ is the four-vector position of the i th particle as a function of the proper time t_i , while $\dot{x}_\mu^i(t_i) = dx_\mu^i/dt_i$ is the velocity four-vector. The first term in the action (3.503) is the ordinary mechanical action in Euclidean space, while the second term defines the electrical interaction of the charges, representing the Maxwell-like field (it is summed over each pair of charges; the factor $\frac{1}{2}$ is to count each pair once, while the term $i = j$ is omitted to avoid self-action; the interaction is a double integral over a delta function of the square of space-time interval I^2 between two points on the paths; thus, interaction occurs only when this interval vanishes, that is, along light cones [WF49]).

Now, from the point of view of Lewinian geometrical force-fields and (loco)motion paths, we can give the following life-space interpretation to the Wheeler-Feynman action (3.503). The mechanical-like locomotion term occurring at the single time t , needs a covariant generalization from the flat 4D Euclidean space to the n D smooth Riemannian manifold, so it becomes (see e.g., [Iva04])

$$S[x] = \frac{1}{2} \int_{t_{ini}}^{t_{fin}} g_{ij} \dot{x}^i \dot{x}^j dt,$$

where g_{ij} is the Riemannian metric tensor that generates the total ‘kinetic energy’ of (loco)motions in the life space.

The second term in (3.503) gives the sophisticated definition of Lewinian force-fields that drive the psychological (loco)motions, if we interpret electrical charges e_i occurring at different times t_i as motivational charges – needs.

Local Micro-Level of LSF_{total} . After having properly defined macro-level MD and CD, with a unique transition map F (including a unique motion path, driving field and smooth geometry), we move down to the *microscopic* LSF-level of rapidly fluctuating MD and CD, where we cannot define a unique and smooth path-field-geometry. The most we can do at this level of *fluctuating uncertainty*, is to formulate an adaptive path integral and calculate overall probability amplitudes for ensembles of local transitions from one LSF-point to the neighboring one. This *probabilistic transition micro-dynamics* functor is defined by a multi-path (field and geometry, respectively) and multi-phase *transition amplitude* $\langle Action|Intention \rangle$ of corresponding to the globally-smooth transition map (3.502). This absolute square of this probability amplitude gives the *transition probability* of occurring the final state of *Action* given the initial state of *Intention*,

$$P(Action|Intention) = |\langle Action|Intention \rangle|^2.$$

The total transition amplitude from the state of *Intention* to the state of *Action* is defined on LSF_{total}

$$\mathcal{TA} \equiv \langle Action|Intention \rangle_{total} : INTENTION_{t_0} \Rightarrow ACTION_{t_1}, \quad (3.504)$$

given by adaptive generalization of the Feynman’s path integral [FH65, Fey72, Fey98]. The transition map (3.504) calculates the *overall probability amplitude*

along a multitude of wildly fluctuating paths, fields and geometries, performing the *microscopic* transition from the micro-state $INTENTION_{t_0}$ occurring at initial micro-time instant t_0 to the micro-state $ACTION_{t_1}$ at some later micro-time instant t_1 , such that all micro-time instants fit inside the global transition interval $t_0, t_1, \dots, t_s \in [t_{ini}, t_{fin}]$. It is symbolically written as

$$\langle Action|Intention \rangle_{total} := \int \mathcal{D}[w\Phi] e^{iS[\Phi]}, \quad (3.505)$$

where the Lebesgue integration is performed over all continuous $\Phi_{con}^i = paths + field + geometries$, while summation is performed over all discrete processes and regional topologies Φ_{dis}^j . The symbolic differential $\mathcal{D}[w\Phi]$ in the general path integral (3.505), represents an *adaptive path measure*, defined as a weighted product

$$\mathcal{D}[w\Phi] = \lim_{N \rightarrow \infty} \prod_{s=1}^N w_s d\Phi_s^i, \quad (i = 1, \dots, n = con + dis), \quad (3.506)$$

which is in practice satisfied with a large N corresponding to infinitesimal temporal division of the four motivational phases (*). Technically, the path integral (3.505) calculates the *amplitude* for the transition functor $\mathcal{TA} : Intention \Rightarrow Action$.

In the exponent of the path integral (3.505) we have the action $S[\Phi]$ and the imaginary unit $i = \sqrt{-1}$ (i can be converted into the real number -1 using the so-called *Wick rotation*, see next subsection).

In this way, we get a range of micro-objects in the local LSF at the short time-level: ensembles of rapidly fluctuating, noisy and crossing paths, force-fields, local geometries with obstacles and topologies with holes. However, by averaging process, both in time and along ensembles of paths, fields and geometries, we recover the corresponding global MD and CD variables.

Infinite-Dimensional Neural Network. The adaptive path integral (3.505) incorporates the *local learning process* according to the standard formula: *New Value = Old Value + Innovation*. The general *weights* $w_s = w_s(t)$ in (3.506) are updated by the *MONITOR* feedback during the transition process, according to one of the two standard neural learning schemes, in which the micro-time level is traversed in discrete steps, i.e., if $t = t_0, t_1, \dots, t_s$ then $t + 1 = t_1, t_2, \dots, t_{s+1}$:

1. A *self-organized, unsupervised* (e.g., Hebbian-like [Heb49]) learning rule:

$$w_s(t+1) = w_s(t) + \frac{\sigma}{\eta} (w_s^d(t) - w_s^a(t)), \quad (3.507)$$

where $\sigma = \sigma(t)$, $\eta = \eta(t)$ denote *signal* and *noise*, respectively, while superscripts d and a denote *desired* and *achieved* micro-states, respectively; or

2. A certain form of a *supervised gradient descent learning*:

$$w_s(t+1) = w_s(t) - \eta \nabla J(t), \quad (3.508)$$

where η is a small constant, called the *step size*, or the *learning rate* and $\nabla J(n)$ denotes the gradient of the ‘performance hyper-surface’ at the t -th iteration.

Both Hebbian and supervised learning are used for the local decision making process (see below) occurring at the intention formation phase $\mathcal{F}$.

In this way, local micro-level of LSF_{total} represents an infinite-dimensional neural network. In the cognitive psychology framework, our adaptive path integral (3.505) can be interpreted as *semantic integration* (see [BF71, Ash94]).

Motion and Decision Making in LSF_{paths}

On the macro-level in the subspace LSF_{paths} we have the (loco)*motion action principle*

$$\delta S[x] = 0,$$

with the *Newtonian-like action* $S[x]$ given by

$$S[x] = \int_{t_{ini}}^{t_{fin}} dt \left[\frac{1}{2} g_{ij} \dot{x}^i \dot{x}^j + \varphi^i(x^i) \right], \quad (3.509)$$

where overdot denotes time derivative, so that $\dot{x}^i$ represents *processing speed*, or (loco)motion velocity vector. The first bracket term in (3.509) represents the kinetic energy T ,

$$T = \frac{1}{2} g_{ij} \dot{x}^i \dot{x}^j,$$

generated by the *Riemannian metric tensor* g_{ij} , while the second bracket term, $\varphi^i(x^i)$, denotes the family of potential force-fields, driving the (loco)motions $x^i = x^i(t)$ (the *strengths* of the fields $\varphi^i(x^i)$ depend on their positions x^i in LSF, see LSF_{fields} below). The corresponding Euler-Lagrangian equation gives the Newtonian-like equation of motion

$$\frac{d}{dt} T_{\dot{x}^i} - T_{x^i} = -\varphi_{x^i}^i, \quad (3.510)$$

(subscripts denote the partial derivatives), which can be put into the standard Lagrangian form

$$\frac{d}{dt} L_{\dot{x}^i} = L_{x^i}, \quad \text{with} \quad L = T - \varphi^i(x^i).$$

In the next subsection we use the micro-level implications of the action $S[x]$ as given by (3.509), for dynamical descriptions of the local decision-making process.

On the micro-level in the subspace LSF_{paths} , instead of a single path defined by the Newtonian-like equation of motion (3.510), we have an ensemble of fluctuating and crossing paths with weighted probabilities (of the unit total sum). This ensemble of micro-paths is defined by the simplest instance of our adaptive path integral (3.505), similar to the Feynman's original *sum over histories*,

$$\langle Action|Intention \rangle_{paths} = \oint \mathcal{D}[wx] e^{iS[x]}, \quad (3.511)$$

where $\mathcal{D}[wx]$ is a functional measure on the *space of all weighted paths*, and the exponential depends on the action $S[x]$ given by (3.509). This procedure can be redefined in a mathematically cleaner way if we Wick-rotate the time variable t to imaginary values $t \mapsto \tau = it$, thereby making all integrals real:

$$\oint \mathcal{D}[wx] e^{iS[x]} \xrightarrow{Wick} \oint \mathcal{D}[wx] e^{-S[x]}. \quad (3.512)$$

Discretization of (3.512) gives the *thermodynamic-like partition function*

$$Z = \sum_j e^{-w_j E^j / T}, \quad (3.513)$$

where E^j is the motion energy eigenvalue (reflecting each possible motivational energetic state), T is the temperature-like environmental control parameter, and the sum runs over all motion energy eigenstates (labeled by the index j). From (3.513), we can further calculate all thermodynamic-like and statistical properties of MD and CD (see e.g., [Fey72]), as for example, *transition entropy* $S = k_B \ln Z$, etc.

From cognitive perspective, our adaptive path integral (3.511) calculates all (alternative) pathways of information flow during the transition *Intention* $\rightarrow$ *Action*.

In the language of transition-propagators, the integral over histories (3.511) can be decomposed into the product of propagators (i.e., Fredholm kernels or Green functions) corresponding to the cascade of the four motivational phases (*)

$$\langle Action|Intention \rangle_{paths} = \oint dx^{\mathcal{F}} dx^{\mathcal{I}} dx^{\mathcal{M}} dx^{\mathcal{T}} K(\mathcal{F}, \mathcal{I}) K(\mathcal{I}, \mathcal{M}) K(\mathcal{M}, \mathcal{T}), \quad (3.514)$$

satisfying the Schrödinger-like equation (see e.g., [Dir82])

$$i \partial_t \langle Action|Intention \rangle_{paths} = H_{Action} \langle Action|Intention \rangle_{paths}, \quad (3.515)$$

where H_{Action} represents the Hamiltonian (total energy) function available at the state of *Action*. Here our 'golden rule' is: the higher the H_{Action} , the lower the microscopic fatigue.

In the connectionist language, our propagator expressions (3.514–3.515) represent *activation dynamics*, to which our *Monitor* process gives a kind of *backpropagation* feedback, a version of the basic supervised learning (3.508).

Mechanisms of Decision-Making under Uncertainty. The basic question about our local decision making process, occurring under uncertainty at the intention formation phase $\mathcal{F}$, is: Which alternative to choose? (see [RBT01, Gro82, Gro99, Gro88, Ash94]). In our path-integral language this reads: Which path (alternative) should be given the highest probability weight w ? Naturally, this problem is iteratively solved by the learning process (3.507–3.508), controlled by the *MONITOR* feedback, which we term *algorithmic approach*.

In addition, here we analyze qualitative mechanics of the local decision making process under uncertainty, as a *heuristic approach*. This qualitative analysis is based on the micro-level interpretation of the Newtonian-like action $S[x]$, given by (3.509) and figuring both processing speed $\dot{x}$ and LTM (i.e., the force-field $\varphi(x)$, see next subsection). Here we consider three different cases:

1. If the potential $\varphi(x)$ is not very dependent upon position $x(t)$, then the more direct paths contribute the most, as longer paths, with higher mean square velocities $[\dot{x}(t)]^2$ make the exponent more negative (after Wick rotation (3.512)).
2. On the other hand, suppose that $\varphi(x)$ does indeed depend on position x . For simplicity, let the potential increase for the larger values of x . Then a direct path does not necessarily give the largest contribution to the overall transition probability, because the integrated value of the potential is higher than over another paths.
3. Finally, consider a path that deviates widely from the direct path. Then $\varphi(x)$ decreases over that path, but at the same time the velocity $\dot{x}$ increases. In this case, we expect that the increased velocity $\dot{x}$ would more than compensate for the decreased potential over the path.

Therefore, the most important path (i.e., the path with the highest weight w) would be one for which any smaller integrated value of the surrounding field potential $\varphi(x)$ is more than compensated for by an increase in kinetic-like energy $\frac{m}{2}\dot{x}^2$. In principle, this is neither the most direct path, nor the longest path, but rather a middle way between the two. Formally, it is the path along which the average Lagrangian is minimal,

$$\langle \frac{m}{2}\dot{x}^2 + \varphi(x) \rangle \longrightarrow \min, \quad (3.516)$$

i.e., the *path that requires minimal memory* (both LTM and WM, see LSF_{fields} below) and *processing speed*. This mechanical result is consistent with the ‘filter theory’ of *selective attention* [Bro58], proposed in an attempt to explain a range of the existing experimental results. This theory postulates a low level filter that allows only a limited number of percepts to reach the brain at any time. In this theory, the importance of conscious, directed attention is minimized. The type of attention involving low level filtering corresponds to the concept of *early selection* [Bro58].

Although we termed this ‘heuristic approach’ in the sense that we can instantly feel both the processing speed $\dot{x}$ and the LTM field $\varphi(x)$ involved, there is clearly a psycho–physical rule in the background, namely the averaging minimum relation (3.516).

From the decision making point of view, all possible paths (alternatives) represent the *consequences* of decision making. They are, by default, *short-term consequences*, as they are modeled in the micro–time–level. However, the path integral formalism allows calculation of the *long-term consequences*, just by extending the integration time, $t_{fin} \rightarrow \infty$. Besides, this *averaging decision mechanics* – choosing the optimal path – actually performs the ‘averaging lift’ in the LSF: from micro– to the macro–level.

Force–Fields and Memory in LSF_{fields}

At the macro–level in the subspace LSF_{fields} we formulate the *force–field action principle*

$$\delta S[\varphi] = 0, \quad (3.517)$$

with the action $S[\varphi]$ dependent on Lewinian force–fields $\varphi^i = \varphi^i(x)$ ($i = 1, \dots, N$), defined as a temporal integral

$$S[\varphi] = \int_{t_{ini}}^{t_{fin}} \mathfrak{L}[\varphi] dt, \quad (3.518)$$

with Lagrangian density given by

$$\mathfrak{L}[\varphi] = \int d^n x \mathcal{L}(\varphi_i, \partial_{x^j} \varphi^i),$$

where the integral is taken over all n coordinates $x^j = x^j(t)$ of the LSF, and $\partial_{x^j} \varphi^i$ are partial derivatives of the field variables over coordinates.

On the micro–level in the subspace LSF_{fields} we have the Feynman–type *sum over fields* φ^i ($i = 1, \dots, N$) given by the adaptive path integral

$$\langle Action | Intention \rangle_{fields} = \oint \mathcal{D}[w\varphi] e^{iS[\varphi]} \xrightarrow{Wick} \oint \mathcal{D}[w\varphi] e^{-S[\varphi]}, \quad (3.519)$$

with action $S[\varphi]$ given by temporal integral (3.518). (Choosing special forms of the force–field action $S[\varphi]$ in (3.519) defines micro–level MD and CD, in the LSF_{fields} space, that is similar to standard quantum–field equations, see e.g., [Ram90].) The corresponding partition function has the form similar to (3.513), but with field energy levels.

Regarding topology of the force fields, we have in place *n–categorical Lagrangian–field structure* on the Riemannian LSF manifold M ,

$$\Phi^i : [0, 1] \rightarrow M, \Phi^i : \Phi_0^i \mapsto \Phi_1^i,$$

generalized from the recursive homotopy dynamics (2.83) in the Appendix, using

$$\begin{aligned} \frac{d}{dt}f_{x^i} &= f_{x^i} \longrightarrow \partial_\mu \left(\frac{\partial \mathcal{L}}{\partial_\mu \Phi^i} \right) = \frac{\partial \mathcal{L}}{\partial \Phi^i}, \\ \text{with } [x_0, x_1] &\longrightarrow [\Phi_0^i, \Phi_1^i]. \end{aligned}$$

Relationship between Memory and Force-Fields. As already mentioned, the subspace LSF_{fields} is related to our *memory storage* [Ash94]. Its global macro-level represents the *long-term memory* (LTM), defined by the least action principle (3.517), related to *cognitive economy* in the model of *semantic memory* [Rat78, CQ69]. Its local micro-level represents *working memory* (WM), a limited-capacity ‘bottleneck’ defined by the adaptive path integral (3.519). According to our formalism, each of Miller’s 7 ± 2 units [Mil56] of the local WM are adaptively stored and averaged to give the global LTM capacity (similar to the physical notion of potential). This averaging memory lift, from WM to LTM represents *retroactive interference*, while the opposite direction, given by the path integral (3.519) itself, represents *proactive interference*. Both retroactive and proactive interferences are examples of the impact of cognitive contexts on memory. Motivational contexts can exert their influence, too. For example, a reduction in task-related recall following the completion of the task is one of the clearest examples of force-field influences on memory: the amount of details remembered of a task declines as the force-field tension to complete the task is reduced by actually completing it.

Once defined, the global LTM potential $\varphi = \varphi(x)$ is then affecting the locomotion transition paths through the path action principle (3.509), as well as general learning (3.507–3.508) and decision making process (3.516).

On the other hand, the two levels of LSF_{fields} fit nicely into the two levels of processing framework, as presented by [CL72], as an alternative to theories of separate stages for sensory, working and long-term memory. According to the *levels of processing framework*, stimulus information is processed at multiple levels simultaneously depending upon its characteristics. In this framework, our macro-level memory field, defined by the fields action principle (3.517), corresponds to the *shallow memory*, while our micro-level memory field, defined by the adaptive path integral (3.519), corresponds to the *deep memory*.

Geometries, Topologies and Noise in LSF_{geom}

On the macro-level in the subspace LSF_{geom} representing an n -dimensional smooth manifold M with the global Riemannian metric tensor g_{ij} , we formulate the *geometrical action principle*

$$\delta S[g_{ij}] = 0,$$

where $S = S[g_{ij}]$ is the n -dimensional *geodesic action* on M ,

$$S[g_{ij}] = \int d^n x \sqrt{g_{ij}} dx^i dx^j. \quad (3.520)$$

The corresponding Euler–Lagrangian equation gives the *geodesic equation* of the *shortest path* in the manifold M ,

$$\ddot{x}^i + \Gamma_{jk}^i \dot{x}^j \dot{x}^k = 0,$$

where the symbol Γ_{jk}^i denotes the so-called *affine connection* which is the source of *curvature*, which is geometrical description for *noise* (see [Ing97, Ing98]). The higher the local curvatures of the LSF-manifold M , the greater the noise in the life space. This noise is the source of our micro-level fluctuations. It can be internal or external; in both cases it curves our micro-LSF.

Otherwise, if instead we choose an n -dimensional Hilbert-like action (see [MTW73]),

$$S[g_{ij}] = \int d^n x \sqrt{\det |g_{ij}|} R, \quad (3.521)$$

where R is the scalar curvature (derived from Γ_{jk}^i), we get the n -dimensional Einstein-like equation: $G_{ij} = 8\pi T_{ij}$, where G_{ij} is the Einstein-like tensor representing geometry of the LSF manifold M (G_{ij} is the trace-reversed Ricci tensor R_{ij} , which is itself the trace of the *Riemann curvature tensor* of the manifold M), while T_{ij} is the n -dimensional *stress-energy-momentum* tensor. This equation explicitly states that *psycho-physics of the LSF is proportional to its geometry*. T_{ij} is important quantity, representing motivational *energy*, geometry-imposed *stress* and *momentum* of (loco)motion. As before, we have our ‘golden rule’: *the greater the T_{ij} -components, the higher the speed of cognitive processes and the lower the macroscopic fatigue*.

The choice between the geodesic action (3.520) and the Hilbert action (3.521) depends on our interpretation of time. If time is not included in the LSF manifold M (non-relativistic approach) then we choose the geodesic action. If time is included in the LSF manifold M (making it a relativistic-like n -dimensional space-time) then the Hilbert action is preferred. The first approach is more related to the information processing and the working memory. The later, space-time approach can be related to the long-term memory: we usually recall events closely associated with the times of their happening.

On the micro-level in the subspace LSF_{geom} we have the adaptive *sum over geometries*, represented by the path integral over all local (regional) Riemannian metrics $g_{ij} = g_{ij}(x)$ varying from point to point on M (modulo diffeomorphisms),

$$\langle Action | Intention \rangle_{geom} = \oint \mathcal{D}[wg_{ij}] e^{iS[g_{ij}]} \xrightarrow{Wick} \oint \mathcal{D}[wg_{ij}] e^{-S[g_{ij}]}, \quad (3.522)$$

where $\mathcal{D}[g_{ij}]$ is diffeomorphism equivalence class of $g_{ij}(x) \in M$.

To include the topological structure (e.g., a number of holes) in M , we can extend (3.522) as

$$\langle Action | Intention \rangle_{geom/top} = \sum_{topol.} \oint \mathcal{D}[wg_{ij}] e^{iS[g_{ij}]}, \quad (3.523)$$

where the topological sum is taken over all connectedness-components of M determined by the *Euler characteristic* χ of M . This type of integral defines the *theory of fluctuating geometries*, a propagator between $(n - 1)$ -dimensional boundaries of the n -dimensional manifold M . One has to contribute a meaning to the integration over geometries. A key ingredient in doing so is to approximate (using simplicial approximation and Regge calculus [MTW73]) in a natural way the smooth structures of the manifold M by piecewise linear structures (mostly using topological simplices Δ). In this way, after the Wick-rotation (3.512), the integral (3.522–3.523) becomes a *simple statistical system*, given by partition function $Z = \sum_{\Delta} \frac{1}{C_{\Delta}} e^{-S_{\Delta}}$, where the summation is over all triangulations Δ of the manifold M , while C_T is the order of the automorphism group of the performed triangulation.

Micro-Level Geometry: the source of noise and stress in LSF. The subspace LSF_{geom} is the source of noise, fluctuations and obstacles, as well as psycho-physical stress. Its micro-level is adaptive, reflecting the human ability to efficiently act within the noisy environment and under the stress conditions. By averaging it produces smooth geometry of certain curvature, which is at the same time the smooth psycho-physics. This macro-level geometry directly affects the memory fields and indirectly affects the (loco)motion transition paths.

The Mental Force Law. As an effective summary of this section, we state that the psychodynamic transition functor $\mathcal{TA} : INTENTION_{t_{ini}} \Rightarrow ACTION_{t_{fin}}$, defined by the generic path integral (3.505), can be interpreted as a *mental force law*, analogous to our musculo-skeletal *covariant force law*, $F_i = mg_{ij}a^j$, and its associated *covariant force functor* $\mathcal{F}_* : TT^*M \rightarrow TTM$ [II05].

Fuzzy Systems

In this Chapter we further develop our fuzzy logic and fuzzy systems modeling approach to fuzzy brain and cognition. Formally, here we are dealing with a *fuzzy brain evolution* 2–functor $\mathcal{E}$, given by

$$\begin{array}{ccc}
 \begin{array}{ccc}
 A & \xrightarrow{f} & B \\
 \downarrow h & \text{CURRENT} & \downarrow g \\
 & \text{FUZZY BRAIN} & \\
 \downarrow & \text{STATE} & \downarrow \\
 C & \xrightarrow{k} & D
 \end{array} & \xrightarrow{\mathcal{E}} & \begin{array}{ccc}
 \mathcal{E}(A) & \xrightarrow{\mathcal{E}(f)} & \mathcal{E}(B) \\
 \downarrow \mathcal{E}(h) & \text{DESIRED} & \downarrow \mathcal{E}(g) \\
 & \text{FUZZY BRAIN} & \\
 \downarrow & \text{STATE} & \downarrow \\
 \mathcal{E}(C) & \xrightarrow{\mathcal{E}(k)} & \mathcal{E}(D)
 \end{array}
 \end{array}$$

Here $\mathcal{E}$ represents an association/projection functor from the *source* 2–category of the current fuzzy brain state, defined as a commutative square of small fuzzy brain categories $A, B, C, D, \dots$ of *current fuzzy brain components* and their causal interrelations $f, g, h, k, \dots$, onto the *target* 2–category of the desired fuzzy brain state, defined as a commutative square of small fuzzy brain categories $\mathcal{E}(A), \mathcal{E}(B), \mathcal{E}(C), \mathcal{E}(D), \dots$ of *evolved fuzzy brain components* and their causal interrelations $\mathcal{E}(f), \mathcal{E}(g), \mathcal{E}(h), \mathcal{E}(k)$. As in the previous section, each causal arrow in above diagram, e.g., $f : A \rightarrow B$, stands for a *generic fuzzy brain dynamorphism*.

4.1 Fuzzy–Logic Control

As the most prominent application of fuzzy–logic systems was in nonlinear control, we start with the fully developed example of a fuzzy–logic nonlinear controller. Recall that the fuzzy controller can be either a direct process controller, or a supervisory controller for several conventional linear PID controllers (see Figure 4.1). This section is intentionally written in “plain English with lots of figures,” to emphasize the well–known fact that the efficient fuzzy–logic control really does not require as a prerequisite a high–level of either

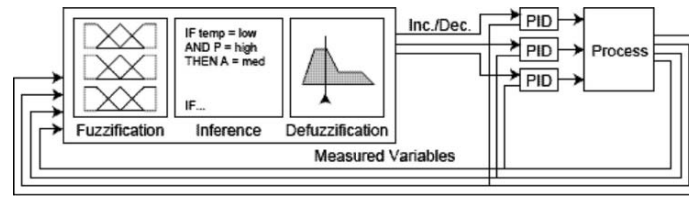


Fig. 4.1. In many applications, keeping a single figure of the process constant is relatively easy and can be facilitated by conventional PID controllers. However, to optimize the operation point of the process, a *supervisory controller* modifies the set points of the PID controllers (adapted from [Alt95]).

mathematical or engineering skills. To make powerful nonlinear dynamical models and controllers using fuzzy-logic, we need just a common sense and a fuzzy software package.

As an example of an efficient fuzzy-logic control, consider the standard crane problem: the container (load) is already picked up from the ship. It must be positioned over the truck, so that the container can be released (see Figure 4.2). Container cranes load and unload containers to and from ships

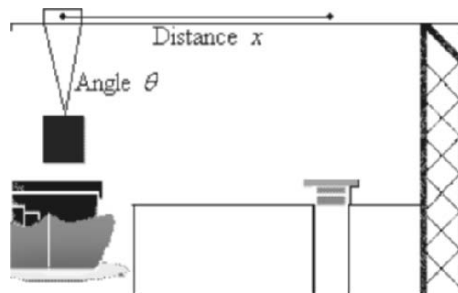


Fig. 4.2. The crane problem: given the distance x and the swaying angle θ , we have to determine the motor power P , so that the time of position the container above the truck is minimal.

in most harbors. They pick up single containers with flexible cables mounted at the crane head. The crane head moves on a horizontal track. When a container is picked up and the crane head starts to move, the container begins to sway. While sway does not affect the transport, a swaying container cannot be released.

There are two trivial solutions to this problem. One is to position the crane head exactly over the target position and wait until the sway dampens to an acceptable level. On a non-windy day, this eventually happens, but it takes far too much time. A container ship has to be loaded and unloaded in minimum time.

The alternative is to move the container so slowly that no sway occurs. Again, this technique works on a non-windy day and takes too much time.

Another solution is to build container cranes where additional cables fix the position of the container during operation. This alternative is very expensive.

Many engineers have attempted to automate this control task via conventional PID, model-based, and fuzzy-logic control strategies.

Conventional PID control was unsuccessful because the control task is inherently nonlinear. For example, sway minimization is important only when the container is close to the target.

Others have tried to derive a mathematical model of the crane to use in a model-based controller. They came up with a fifth-degree differential equation that describes the mechanical behavior. In theory, a controller design based on this model works. In reality, it does not.

One reason for failure is that the weight of the container is unknown. Also, the crane-motor behavior is not as linear as assumed in the model. Its gear box involves slack, its head only moves with friction, and its cables involve elasticity. As well, disturbances such as wind gusts are not included in the model.

On the other hand, a human operator can control a crane without differential equations. An operator does not even use the cable-length sensors that a model-based solution requires. Once the container is picked up, the operator starts the crane with medium motor power to see how the container sways.

Depending on the reaction, the motor power is adjusted to get the container a little behind the crane head. In this position, maximum speed is reached with minimum sway.

In approaching the target position, the operator reduces motor power or applies negative power. The container gets a little ahead of the crane head until the container almost reaches target position.

Motor power is then increased so the crane head is over target position and sway is zero. No differential equations are required, and disturbances and nonlinearities are compensated by the operator's observation of the container's position.

The operator's control strategy can be described by several rules:

(i) start with medium power; (ii) if you're still far away from the target, adjust the motor power so the container gets a little behind the crane head; (iii) if you're closer to the target, reduce speed so the container gets a little ahead of the crane head (iv) when the container is very close to target position, power up the motor; (v) when the container is over the target and the sway is zero, stop the motor [Alt95].

The advantage of fuzzy logic is that it can use the same sort of rules as the human operator.

To automate control of this crane, sensors are used for the head position (Distance x) and the angle of the container sway (Angle θ). Using these inputs to describe the current condition of the crane, the rules can be translated into the fuzzy IF-THEN statements. Where, the first condition describes the value

No.	Angle	Distance	Power
1	PS	0	NS
2	0	0	0
3	PS	Cl	NS
4	0	Cl	0
5	NS	Cl	PS
6	NS	Md	PB
7	NB	MD	PS
8	0	FR	PS
9	NS	FR	PB

Table 4.1. Fuzzy IF–THEN rule–base for the crane controller.

of Distance, and the second the value of Angle. The conditions are combined by AND since both have to be valid for the respective situation. The full fuzzy–rule base is given in Table 4.1.

Figure 4.3 shows the complete structure of a fuzzy–logic crane controller. All sensor signals have to be translated into linguistic variables. A measured distance of 12m has to be translated to the linguistic value ‘still medium, just slightly far’. This step is called fuzzification because it uses fuzzy sets for translating real variables into linguistic variables. Once all input variable values are translated into linguistic variable values, the so–called fuzzy–inference step evaluates the IF–THEN fuzzy rules that define system behavior. This step yields a linguistic value for the linguistic variable. So, the linguistic result for Power could be ‘a little less than medium’. Finally, defuzzification translates this linguistic result into a real value that represents the power setting of the motor in kilowatts.

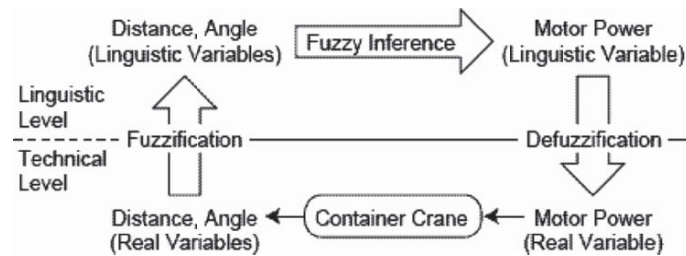


Fig. 4.3. The complete control loop of the fuzzy–logic crane controller has three steps. The variables Distance and Angle measured at the crane are translated into linguistic variables (fuzzification) and used to evaluate the condition of the fuzzy rules (fuzzy inference). A linguistic value for Power is translated back into a numerical value (defuzzification) (adapted from [Alt95]).

Recall that the development of fuzzy–logic systems involves specific design steps:

(i) Design the inference structure, specify how the output variables connect to the input variables by the rule blocks. (ii) Define the linguistic variables, the variables form the vocabulary used by the fuzzy–logic rules expressing the control strategy. (iii) Create an initial fuzzy–logic rule base using all available knowledge on how the system should perform. (iv) Debug, test, and verify the system off–line for completeness and non–ambiguity, use a software simulation or sample data of the process if it exists in this step. (v) Debug online, connect the fuzzy–logic system to the process under control and analyze its performance in operation. Because fuzzy logic lets you modify the system in a straightforward way from the performance you observe, this step can expedite system design rapidly.

For example, the above $2 \rightarrow 1$ crane controller structure implemented in *fuzzyTechTM* is depicted in Figure 4.4.

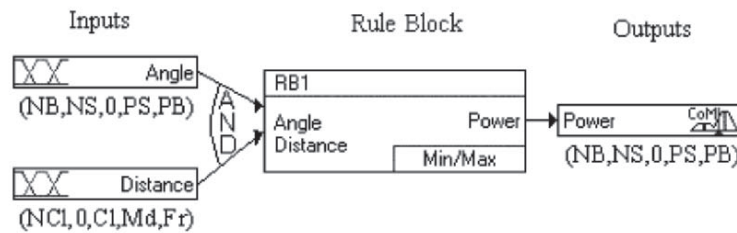


Fig. 4.4. Fuzzy inference IN–OUT diagram for the crane controller, as implemented in *fuzzyTechTM*.

The fuzzy membership functions for distance (x), angle (θ) and output power (P), together with the corresponding time plots and the fuzzy control surface for the crane controller are given in Figure 4.5.

On a standard 12MHz 8051 micro–controller (MCU), small fuzzy–logic systems compute in just one millisecond, and 16–bit MCUs can compute large fuzzy–logic systems in the same amount of time. This speed enables the integration of a fuzzy–logic system with most embedded system designs. However, some embedded system applications require even faster computation (e.g., anti–lock brakes in cars, ignition control, or hard–drive positioning). To expedite the fuzzy–logic algorithm, some semiconductor manufacturers include specific fuzzy–logic instructions with their new–generation MCUs. For example, Motorola’s 68HC12 MCU features a complete fuzzy–logic function set in assembly language.

A slightly finer $2 \rightarrow 1$ nonlinear dynamical system would have a generic structure as depicted in Figure 4.6.

At the end of this section, we present several visual examples of fuzzy dynamical models simulated using *fuzzyTechTM*:

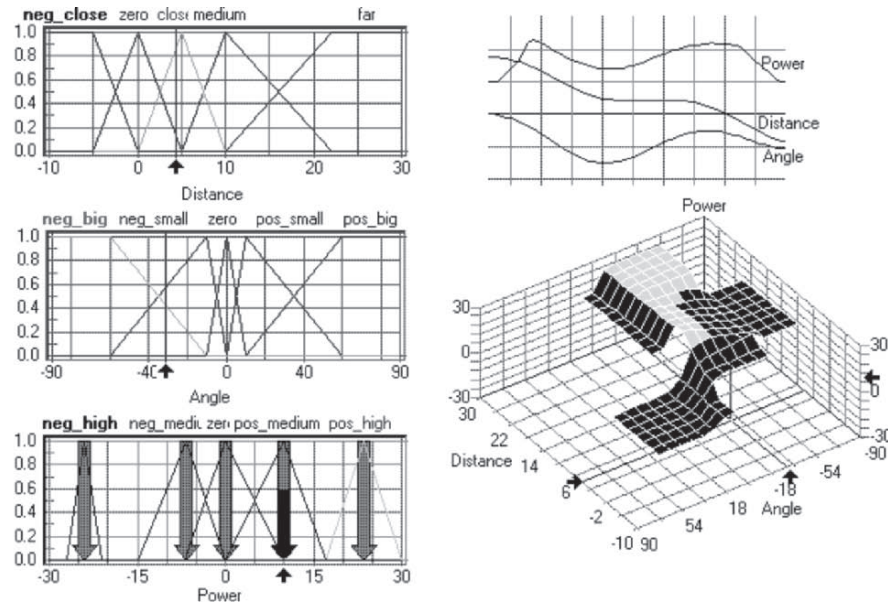


Fig. 4.5. Fuzzy membership functions (left), time plots (right-up), and control surface (right-down) for the crane controller (simulated using *fuzzyTechTM*).

- Tire Comfort Evaluation (Figure 4.7);
- Customer Selection for Direct Marketing (Figure 4.8);
- Fuzzy Trend Identification (Figure 4.9);
- Applicant Assessment for Mortgage Loan (Figure 4.10); and
- Fuzzy Car Autopilot (Figure 4.11).

4.2 Fuzzy Cognitive Maps

Recall that the notion of a *fuzzy cognitive map* (FCM) was introduced by Bart Kosko in [Kos86]. FCMs have their roots in graph theory, first formulated by Leonard Euler in 1736. This method is simple and effective for analyzing unsupervised data by directed graphs and their adjacency matrices. FCMs are based on the opinion of experts. FCMs model the world as a collection of concepts and causal relations between those concepts. FCMs are useful in symbolic representations of semantic networks and they also provide a method for modeling cognitive and empirical cause-effect processes. Their advantage lies in that they allow feedback processes. Kosko [Kos86] coined the term FCM when he modified Axelrod's cognitive maps [Axe76] by applying fuzzy causal functions with real numbers in $[-1, 1]$ to connections (also called lines, edges,

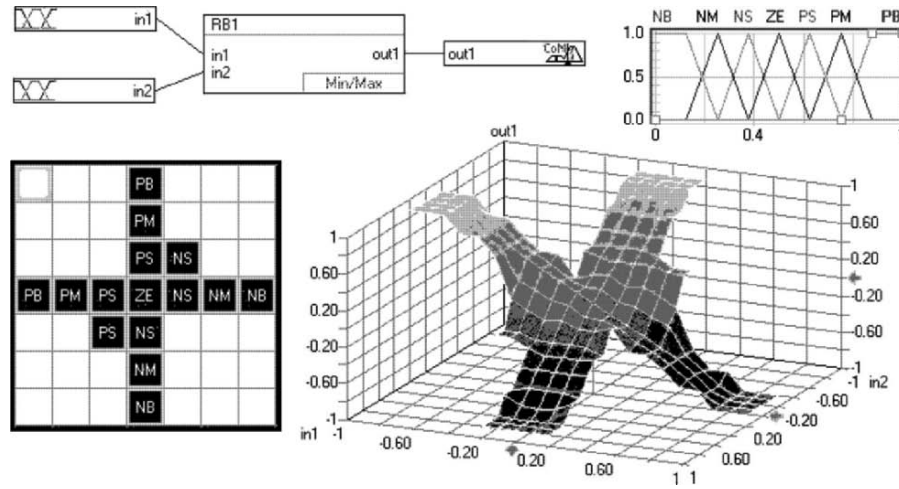


Fig. 4.6. A generic structure of a nonlinear dynamical system with two inputs and one output, each partitioned by seven triangular fuzzy membership functions, together with its rule-base given in the form of a FAM matrix and the associated control surface (reflecting the shape of the FAM matrix).

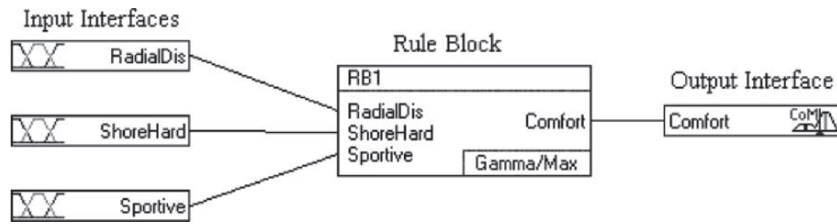


Fig. 4.7. Fuzzy inference In-Out diagram for tire comfort evaluation.

synapses). The basis of FCMs is auto-associative neural network architecture [Rei98].

4.2.1 FCM Definitions

An FCM is a directed graph with concepts like policies, events etc. as nodes and causalities as edges. It represents causal relationship between concepts. If increase (or decrease) in one concept leads to increase (or decrease) in another one, we associate the edge weight 1 to the causal relationship (arrow) between them. If there exists no relation between two concepts the value 0 is given to the arrow. If increase (or decrease) in one concept decreases (or increases) another, then we give the value -1 to the arrow between them.

When the nodes of the FCM are fuzzy sets, they are called fuzzy nodes.

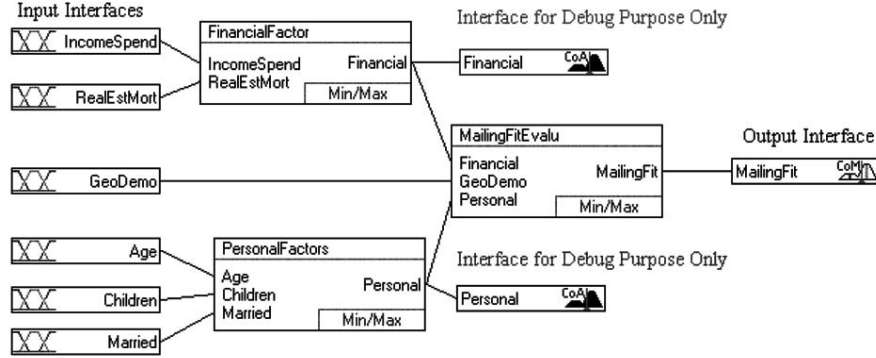


Fig. 4.8. Fuzzy inference In-Out diagram for the customer selection for direct marketing.

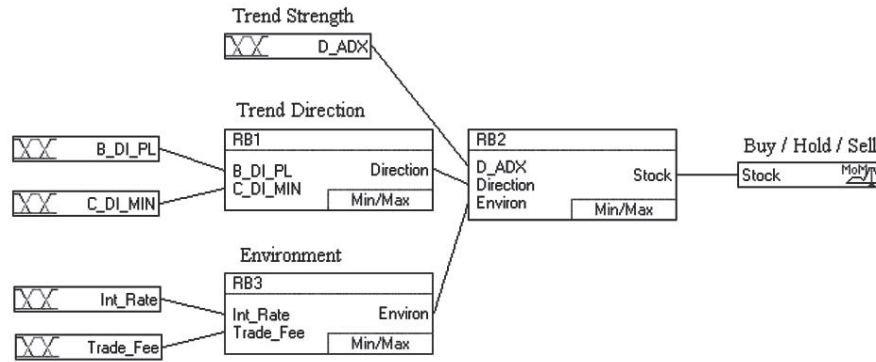


Fig. 4.9. Fuzzy inference In-Out diagram for the fuzzy trend identification.

FCMs with edge weights or causalities from the set $\{-1, 0, 1\}$ are called simple FCMs.

Consider the nodes / concepts $C_1, \dots, C_n$ of an FCM. Suppose a directed graph is drawn using edge weight $e_{ij} \in \{0, 1, -1\}$. The matrix E defined by $E = (e_{ij})$ where e_{ij} is the weight of the directed edge $C_i C_j$. E is called the *adjacency matrix* of the FCM, also known as the *connection matrix* of the FCM.¹

Let $C_1, C_2, \dots, C_n$ be the nodes of an FCM. $A = (a_1, a_2, \dots, a_n)$ where $a_i \in \{0, 1\}$. A is called the *instantaneous state vector* and it denotes the on-off position of the node at an instant. We have

¹ Note that all matrices associated with an FCM are always square matrices with zero diagonal entries.

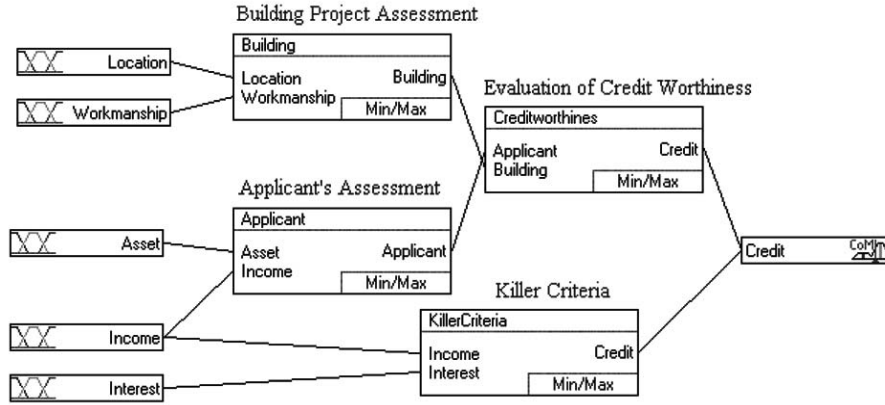


Fig. 4.10. Fuzzy inference In-Out diagram for applicant assessment for the mortgage loan.

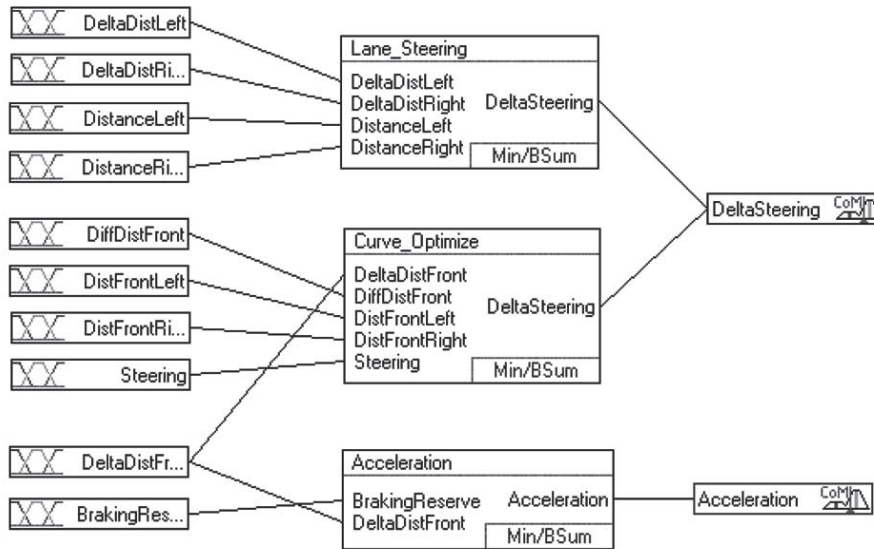


Fig. 4.11. Fuzzy inference In-Out diagram for the fuzzy car autopilot.

$$a_i = \begin{cases} 0 & \text{if } a_i \text{ is off} \\ 1 & \text{if } a_i \text{ is on} \end{cases}, \quad \text{for } i = 1, 2, \dots, n.$$

Let $C_1, C_2, \dots, C_n$ be the nodes of an FCM. Let $\overline{C_1 C_2}, \overline{C_2 C_3}, \overline{C_3 C_4}$, be the edges of the FCM ($i \neq j$). Then the edges form a *directed cycle*. We have a *cyclic FCM* if it possesses a directed cycle. We have an *acyclic FCM* if it does not possess any directed cycle.

An FCM with cycles is said to have a *feedback*. When there is a feedback in an FCM, i.e., when the causal relations flow through a cycle in a revolutionary way, the FCM represents a *dynamical system*.

Let $\overline{C_1 C_2}, \overline{C_2 C_3}, \dots, \overline{C_{n-1} C_n}$ be a cycle. When C_i is switched on and if the causality flows through the edges of a cycle and if it again causes C , we say that the FCM–dynamical system loops around. This is true for any node C , for $i = 1, 2, \dots, n$. The equilibrium state for this dynamical system is called the *hidden pattern*.

If the equilibrium state of an FCM–dynamical system is a unique state vector, then it is called a *fixed-point*.

For example, consider an FCM with nodes $C_1, C_2, \dots, C_n$. Let us start the FCM–dynamical system by switching on C_1 . Let us assume that the FCM settles down with C_1 and C_n on, i.e., the state vector remains as $(1, 0, 0, \dots, 0, 1)$ this state vector is called the fixed-point.

If the FCM settles down with a state vector repeating in the form

$$A_1 \rightarrow A_2 \rightarrow \dots \rightarrow A_i \rightarrow A_1$$

then this equilibrium is called a *limit cycle*.

Finite number of FCMs can be combined together to produce the joint effect of all the FCMs. Let $E_1, E_2, \dots, E_p$ be the adjacency matrices of the FCMs with nodes $C_1, C_2, \dots, C_n$. Then the *combined FCM* is obtained by adding all the adjacency matrices $E_1, E_2, \dots, E_p$. We denote the combined FCM adjacency matrix by $E = E_1 + E_2 + \dots + E_p$.

FCMs have several advantages as well as some disadvantages. The main advantage of this method it is simple. It functions on expert's opinion. When the data happens to be an unsupervised one the FCM comes handy. This is the only known fuzzy technique that gives the hidden pattern of the situation. As we have a very well known theory, which states that the strength of the data depends on, the number of experts' opinion we can use combined FCMs with several experts' opinions.

At the same time the disadvantage of the combined FCM is when the weights are 1 and -1 for the same $C_i C_j$, when we have the sum adding to zero and thus at all times the connection matrices $E_1, \dots, E_k$ may not be conformable for addition.

Combined conflicting opinions tend to cancel out and assisted by the strong law of large numbers, a consensus emerges as the sample opinion approximates the underlying population opinion. This problem would be easily overcome if the FCM entries are only 0 and 1. For details, see [Kos86].

4.2.2 FCM Properties and Models

As defined above, FCMs are fuzzy signed directed graphs with feedback. The directed edge e_{ij} from causal concept C_i to concept C_j measures how much

C_i causes C_j . The time varying concept function $C_i(t)$ measures the non-negative occurrence of some fuzzy event, perhaps the strength of a political sentiment, historical trend or military objective [KS03].

FCMs are used to model several types of problems varying from gastric-appetite behavior to popular political developments. FCMs are also used to model plant control in robotics.

The edges e_{ij} take values in the fuzzy causal interval $[-1, 1]$. $e_{ij} = 0$ indicates no causality, $e_{ij} > 0$ indicates causal increase or positive causality: C_j increases as C_i increases (or C_j decreases as C_i decreases). $e_{ij} < 0$ indicates causal decrease or negative causality: C_j decreases as C_i increases (or C_j increases as C_i decreases). Simple FCMs have edge values in $-1, 0, 1$. Then if causality occurs, it occurs to a maximal positive or negative degree. Simple FCMs provide a quick first approximation to an expert stand or printed causal knowledge.

We illustrate this by the following example of a *socio-economic model*, constructed with Population, Crime, Economic condition, Poverty and Unemployment as nodes or concepts. Here the simple trivalent directed graph is given in Figure 4.12, reflecting the experts opinion. Causal feedback loops abound in FCMs in thick tangles. Feedback prevents the graph-search techniques as used in AI expert systems.

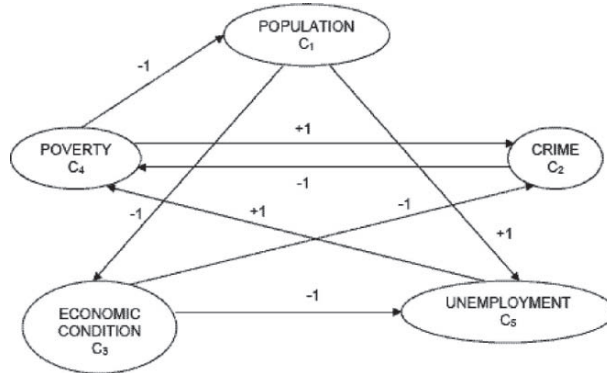


Fig. 4.12. A FCM representing a socio-economic model (adapted from [KS03]).

FCMs feedbacks allow experts to freely draw causal pictures of their problems and also allow causal adaptation laws, inferring causal links from simple data. FCM feedback force us to abandon graph search, forward and especially backward chaining, as in AI expert systems. Instead we view the FCM as a dynamical system and take its equilibrium behavior as a forward-evolved inference. Synchronous FCMs behave as *temporal associative memories* (TAMs). We can always, in case of a model, add two or more FCMs to produce a new

FCM. The strong law of large numbers ensures in some sense that knowledge reliability increases with expert sample size.

We can reason with FCMs. We can pass state vectors C repeatedly through the FCM connection matrix E , thresholding or nonlinearly transforming the result after each pass. Independent of the FCMs size, it quickly settles down to a temporal associative memory limit cycle or fixed-point which is the hidden pattern of the system for that state vector C . The limit cycle or fixed-point inference summarizes the joint effects of all the interacting fuzzy concepts.

For example, consider the 5×5 causal connection matrix E that represents the socio-economic FCM given in figure in Figure 4.12:

$$E = \begin{bmatrix} 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 & -1 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

Concept nodes can represent processes, events, values or policies. Consider the first node $C_1 = 1$. We hold or clamp C_1 on the TAM recall process. Threshold signal functions synchronously update each concept after each pass, through the connection matrix E . We start with the population $C_1 = (1 \ 0 \ 0 \ 0 \ 0)$. If the arrow indicates the threshold operation, we have:

$$\begin{aligned} C_1 E &= (0 \ 0 \ -1 \ 0 \ 1) \rightarrow (1 \ 0 \ 0 \ 0 \ 1) = C_2, \\ C_2 E &= (0 \ 0 \ -1 \ 1 \ 1) \rightarrow (1 \ 0 \ 0 \ 1 \ 1) = C_3, \\ C_3 E &= (-1 \ 1 \ -1 \ 1 \ 1) \rightarrow (1 \ 1 \ 0 \ 1 \ 1) = C_4, \\ C_4 E &= (-1 \ 1 \ -1 \ 0 \ 1) \rightarrow (1 \ 1 \ 0 \ 0 \ 1) = C_5, \\ C_5 E &= (0 \ 0 \ -1 \ 0 \ 1) \rightarrow (1 \ 0 \ 0 \ 0 \ 1) = C_6 = C_2, \end{aligned}$$

So the increase in population results in the unemployment problem, which is a limit cycle [KS03].

This example illustrates the strengths and weaknesses of FCM analysis. FCM allows experts to represent factual and evaluative concepts in an interactive framework. Experts can quickly draw FCM pictures or respond to questionnaires. Experts can consent or dissent to the local causal structure and perhaps the global equilibrium. The FCM knowledge representation and inferencing structure reduces to simple vector-matrix operations, favors integrated circuit implementation and allows extension to neural statistical or dynamical systems techniques. Yet an FCM equally encodes the experts' knowledge or ignorance, wisdom or prejudice. Worse, different experts differ in how they assign causal strengths to edges and in which concepts they deem causally relevant. The FCM seems merely to encode its designers' biases and may not even encode them accurately.

FCM combination provides a partial solution to this problem. We can additively superimpose each experts FCM in associative memory fashion, even

though the FCM connection matrices $E_1, \dots, E_K$ may not be conformable for addition. Combined conflicting opinions tend to cancel out and assisted the strong law of large numbers a consensus emerges as the sample opinion approximates the underlying population opinion. FCM combination allows knowledge researchers to construct FCMs with iterative interviews or questionnaire mailings.

The laws of large numbers require that the random samples be independent identically distributed random variables with finite variance. Independence models each experts individually. Identical distribution models a particular domain focus.

We combine arbitrary FCM connection matrices $F_1, F_2, \dots, F_K$ by adding augmented FCM matrices. Each augmented matrix F_i has n rows and columns, where n equals the total number of distinct concepts used by the experts. We permute the rows and columns of the augmented matrices to bring them into mutual coincidence. Then we add the F_i point wise to yield the *combined FCM matrix*: $F = \sum_i F_i$. We can then use F to construct the combined FCM digraph.

Even if each expert gives trivalent description in $\{-1, 0, 1\}$, the combined (and normalized) FCM entry f_{ij} tends to be in $\{-1, 1\}$. The strong law of large numbers ensures that f_{ij} provides a rational approximation to the underlying unknown population opinion of how much C_i affects C_j . We can normalize f_{ij} by the number K of experts. Experts tend to give trivalent evaluations more readily and more accurately than they give weighted evaluations. When transcribing interviews or documents, a knowledge engineer can more reliably determine an edge's sign than its magnitude.

Some experts may be more credible than others. We can weight each expert with nonnegative credibility weight, weighing the *augmented FCM matrix*: $F = \sum_i w_i F_i$.

The weights need not be in $[0, 1]$; the only condition is they should be non-negative. Different weights may produce different equilibrium limit cycles or fixed-points as hidden patterns. We can also weigh separately any submatrix of each experts augmented FCM matrix.

Augmented FCM matrices imply that every expert causally discusses every concept $C_1, \dots, C_n$. If an expert does not include C_j in his FCM model the expert implicitly say that C_j is not causally relevant. So the j th row and the j th column of his augmented connection matrix contains only zeros [KS03].

4.3 Takagi–Sugeno Fuzzy Dynamics

4.3.1 The Basic Takagi–Sugeno Model

As is first proposed in [TS85] a nonlinear system can be described based on fuzzy approach in the following procedure [ZWL02]. First, the system is

decomposed into simple linear systems through fuzzy implications (rules) of the following form:

$$\begin{aligned} \text{Plant Rule } i \text{ } (= 1, \dots, r) : & \text{ IF } \theta_1 \text{ is } \mu_{i1} \text{ and } \dots \theta_p \text{ is } \mu_{ip} \\ & \text{ THEN } \dot{x}(t) = A_i x(t) + B_i u(t), \end{aligned}$$

where $x(t) \in R^n$ is the vector of state variables, $u(t) \in R^m$ is the vector of control inputs, A_i and B_i are constant matrices of appropriate dimensions, $\theta_j(x) (j = 1, \dots, p)$ are the premise variables, which are the functions of state variables, $\mu_{ij} (i = 1, \dots, r; j = 1, \dots, p)$ are the fuzzy sets, r is the number of the IF-THEN rules, and p is the number of the premise variables. It is assumed that the premise variables are independent of the input variables $u(t)$.

Second, the overall fuzzy model is achieved by fuzzy blending (aggregation) of each individual rule (model) as follows:

$$\dot{x}(t) = \frac{\sum_{i=1}^r \omega_i(\theta) [A_i x(t) + B_i u(t)]}{\sum_{i=1}^r \omega_i(\theta)}, \quad (4.1)$$

where $\theta = [\theta_1, \dots, \theta_p]$ and $\omega_i : R^p \rightarrow [0, 1]$, $i = 1, \dots, r$, are the membership function of the systems belonging to plant rule i .

Let us define

$$\alpha_i(\theta) = \frac{\omega_i(\theta)}{\sum_{i=1}^r \omega_i(\theta)}.$$

System (4.1) admits the following form:

$$\dot{x}(t) = \sum_{i=1}^r \alpha_i(\theta) [A_i x(t) + B_i u(t)], \quad (4.2)$$

with the constraints

$$\alpha_i(\theta) \geq 0, \quad \sum_{i=1}^r \alpha_i(\theta) = 1.$$

4.3.2 Local and Global Fuzzy Dynamical Models

During the past two decades, many systematic fuzzy controller design methods have been developed based on the *Takagi-Sugeno (TS) fuzzy model* [TS85], or the fuzzy dynamic models. The basic idea of these methods is: (i) to represent the complex nonlinear system in a family of local linear models, each linear model represents the dynamics of the complex system in one local region; (ii) to construct a global nonlinear model by aggregating all the local models through the fuzzy membership functions. The primary advantage of this model is that the controller design can be mainly based on each local model, which is much easier than that for nonlinear systems in the global region, and then the global controller can be constructed from the local controllers.

Many physical systems are very complex in practice so that their rigorous mathematical models can be very difficult to get if not impossible. However, many physical systems can indeed be expressed in some form of mathematical models locally, or those systems can be expressed as an aggregation of a set of mathematical models. Various fuzzy models have been proposed in the last decade [TS85]. Feng [Fen02] considered using the following fuzzy model to represent a complex single-input single-output (SISO) continuous-time system that includes both fuzzy inference rules and local analytic linear models (in the state-space form):

$$\begin{aligned}
 R^i : & \text{ IF } z_1 \text{ is } F_1^i \text{ AND } \dots z_s \text{ is } F_s^i \quad (i = 1, \dots, m) \\
 & \text{ THEN } \dot{x}(t) = A_i x(t) + B_i u(t), \quad y = C_i x(t), \quad \text{where}
 \end{aligned} \tag{4.3}$$

$$\begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \dots & 1 \\ a_{i1} & a_{i2} & a_{i3} & \dots & a_{in} \end{bmatrix}, \quad B_i = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ b_i \end{bmatrix}, \quad C_i = [1 \ 0 \ \dots \ 0].$$

The corresponding discrete-time TS-model reads:

$$\begin{aligned}
 R^i : & \text{ IF } z_1 \text{ is } F_1^i \text{ AND } \dots z_s \text{ is } F_s^i \quad (i = 1, \dots, m) \\
 & \text{ THEN } x(t+1) = A_i x(t) + B_i u(t), \quad y = C_i x(t).
 \end{aligned}$$

Let $\mu(z(t))$ be the normalized membership function of the inferred fuzzy set F^i in (4.3) where

$$F^i = \bigcap_{j=1}^s F_s^i \quad \text{and} \quad \sum_{i=1}^m = 1.$$

The model (4.3) only represents the properties of the system in a local region and thus is referred to as the *local fuzzy dynamical model*.

Now, by using a standard fuzzy inference method, that is, using a singleton fuzzifier, product fuzzy inference and center-average defuzzifier, the following *global fuzzy dynamical model* can be obtained (in the state-space form) [Fen02]:

$$\dot{x} = \mu_i A_i x + \mu_i B_i u, \quad y = \mu_i C_i x.$$

4.3.3 Affine Fuzzy System

The continuous *affine fuzzy system* proposed in [KKH03] has the following IF–THEN form :

$$\begin{aligned}
 R^i : & \text{ IF } x_1 \text{ is } M_1^i \text{ AND } \dots x_n \text{ is } M_n^i \\
 & \text{ THEN } \dot{x} = A_i x + \mu_i,
 \end{aligned} \tag{4.4}$$

where $x^T = (x_1 x_2 \dots x_n)$; R^i ($i = 1, \dots, r$) denotes the i th fuzzy rule and $M_1^i \dots M_n^i$ are fuzzy variables. In (4.4), it can be noted that the local subsystem includes a constant affine term. The input-output form of the fuzzy system of (4.4) is represented as

$$\dot{x} = \frac{\omega_i (A_i x + \mu_i)}{\sum_{i=1}^r \omega_i} = h_i(x) [A_i x + \mu_i],$$

where

$$\begin{aligned} \omega_i(x) &= \prod_{j=1}^n M_j^i(x_j), & h_i(x) &= \frac{\omega_i(x)}{\sum_{i=1}^r \omega_i(x)}, \\ h_i(x) &\geq 0, & \sum_{i=1}^r h_i(x) &= 1. \end{aligned}$$

4.3.4 Fuzzy Modeling and Control of Chaotic Systems

TS-Modeling of Classical Chaotic Systems

The Takagi-Sugeno fuzzy model is a universal approximator for nonlinear dynamical systems. Following [TW01], we present in the TS modeling framework several typical chaotic systems with the control input term added.

Forced Lorenz Equations

Consider the standard Lorenz equations with the forcing (input) term:

$$\begin{aligned} \dot{x}_1(t) &= -ax_1(t) + ax_2(t) + u(t), \\ \dot{x}_2(t) &= cx_1(t) - x_2(t) - x_1(t)x_3(t), \\ \dot{x}_3(t) &= x_1(t)x_2(t) - bx_3(t), \end{aligned}$$

where a, b, c are constants and $u(t)$ is the input term. Assume that $x_1(t) \in [-d, d]$ and $d > 0$. Then, we can have the following fuzzy model which exactly represents the nonlinear equation under $x_1(t) \in [-d, d]$:

$$\begin{aligned} \text{Rule 1 :} & \quad IF \quad x_1(t) \quad is \quad M_1, \\ \text{THEN} & \quad \dot{x}(t) = A_1 x(t) + Bu(t); \\ \text{Rule 2 :} & \quad IF \quad x_1(t) \quad is \quad M_2, \\ \text{THEN} & \quad \dot{x}(t) = A_2 x(t) + Bu(t); \end{aligned}$$

where

$$\begin{aligned} x(t) &= [x_1(t) \quad x_2(t) \quad x_3(t)]^T, \\ A_1 &= \begin{bmatrix} -a & a & 0 \\ c & -1 & -d \\ 0 & d & -b \end{bmatrix}, & A_2 &= \begin{bmatrix} -a & a & 0 \\ c & -1 & d \\ 0 & -d & -b \end{bmatrix}, & B &= \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \\ M_1(x_1(t)) &= \frac{1}{2} \left(1 + \frac{x_1(t)}{d} \right), & M_2(x_1(t)) &= \frac{1}{2} \left(1 - \frac{x_1(t)}{d} \right). \end{aligned}$$

Forced Duffing Oscillator

Consider the forced Duffing oscillator:

$$\begin{aligned}\dot{x}_1(t) &= x_2(t), \\ \dot{x}_2(t) &= -x_1^3(t) - ax_2(t) + b \cos(t) + u(t).\end{aligned}$$

where a, b are constants and $u(t)$ is the input term. Assume that $x_1(t) \in [-d, d]$ and $d > 0$. Then we can have the following fuzzy model (similar to the Lorenz one above):

$$\begin{aligned}\text{Rule 1 :} \quad & IF \quad x_1(t) \quad is \quad M_1, \\ \text{THEN} \quad & \dot{x}(t) = A_1 x(t) + Bu^*(t); \\ \text{Rule 2 :} \quad & IF \quad x_1(t) \quad is \quad M_2, \\ \text{THEN} \quad & \dot{x}(t) = A_2 x(t) + Bu^*(t); \\ \text{where} \\ x(t) &= [x_1(t) \quad x_2(t)]^T, \quad u^*(t) = u(t) + b \cos(t) \\ A_1 &= \begin{bmatrix} 0 & -1 \\ 0 & a \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & 1 \\ -d^2 & a \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \\ M_1(x_1(t)) &= 1 - \frac{x_1^2(t)}{d^2}, \quad M_2(x_1(t)) = \frac{x_1^2(t)}{d^2}.\end{aligned}$$

Forced Rossler Equations

Consider the forced Rossler equations:

$$\begin{aligned}\dot{x}_1(t) &= -x_2(t) - x_3(t), \\ \dot{x}_2(t) &= x_1(t) + ax_2(t), \\ \dot{x}_3(t) &= bx_1(t) - [c - x_1(t)]x_3(t) + u(t),\end{aligned}$$

where a, b, c are constants and $u(t)$ is the input term. Assume that $x_1(t) \in [-d, d]$ and $d > 0$. Then, we can have the following fuzzy model which exactly represents the nonlinear equation under $x_1(t) \in [-d, d]$:

$$\begin{aligned}\text{Rule 1 :} \quad & IF \quad x_1(t) \quad is \quad M_1, \\ \text{THEN} \quad & \dot{x}(t) = A_1 x(t) + Bu(t); \\ \text{Rule 2 :} \quad & IF \quad x_1(t) \quad is \quad M_2, \\ \text{THEN} \quad & \dot{x}(t) = A_2 x(t) + Bu(t); \\ \text{where} \\ x(t) &= [x_1(t) \quad x_2(t) \quad x_3(t)]^T, \\ A_1 &= \begin{bmatrix} 0 & -1 & -1 \\ 1 & a & 0 \\ b & 0 & -d \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & -1 & -1 \\ 1 & a & 0 \\ b & 0 & d \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \\ M_1(x_1(t)) &= \frac{1}{2} \left(1 + \frac{c - x_1(t)}{d} \right), \quad M_2(x_1(t)) = \frac{1}{2} \left(1 - \frac{c - x_1(t)}{d} \right).\end{aligned}$$

The fuzzy models above have a single input. We can also consider the multi-input case. For instance, we may consider Lorenz equations with multi-inputs:

$$\begin{aligned}\dot{x}_1(t) &= -ax_1(t) + ax_2(t) + u_1(t), \\ \dot{x}_2(t) &= cx_1(t) - x_2(t) - x_1(t)x_3(t) + u_2(t), \\ \dot{x}_3(t) &= x_1(t)x_2(t) - bx_3(t) + u_2(t).\end{aligned}$$

As before, we can derive the following fuzzy model to exactly represent the nonlinear ODEs under $x_1(t) \in [-d, d]$:

$$\begin{aligned}\text{Rule 1 :} \quad & IF \quad x_1(t) \quad is \quad M_1, \\ \text{THEN} \quad & \dot{x}(t) = A_1x(t) + Bu(t); \\ \text{Rule 2 :} \quad & IF \quad x_1(t) \quad is \quad M_2, \\ \text{THEN} \quad & \dot{x}(t) = A_2x(t) + Bu(t); \\ & \text{where} \\ & x(t) = [x_1(t) \quad x_2(t) \quad x_3(t)]^T, \quad u(t) = [u_1(t) \quad u_2(t) \quad u_3(t)]^T, \\ & A_1 = \begin{bmatrix} -a & a & 0 \\ c & -1 & -d \\ 0 & d & -b \end{bmatrix}, \quad A_2 = \begin{bmatrix} -a & a & 0 \\ c & -1 & d \\ 0 & -d & -b \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \\ & M_1(x_1(t)) = \frac{1}{2} \left(1 + \frac{x_1(t)}{d} \right), \quad M_2(x_1(t)) = \frac{1}{2} \left(1 - \frac{x_1(t)}{d} \right).\end{aligned}$$

OGY–Principles of Chaos Control

Recall that according to OGY [OGY90], the simplest way to control chaotic behavior is to adjust the parameters of a system to a regime of periodic rather than chaotic oscillation. This can be quite easily done by means of fuzzy–logic robust control. For this, we can use fuzzy–implication rules, like:

- IF d_{KY} is very large, OR h_{KS} is very large, THEN σ_i is very large AND/OR σ_j is very small.
- IF d_{KY} is very small, OR h_{KS} is very small, THEN σ_i is very small AND/OR σ_j is very large.

Here $\sigma_{i,j}$ denote two sample control parameters (e.g., inertia moments of body segments with respect to different Euler axes of rotation), while d_{KY} and h_{KS} are *Kaplan–Yorke dimension* and *Kolmogorov–Sinai entropy*, respectively. For these rules we can assign appropriate nonlinear membership functions $\sigma_{i,j}(d_{KY})$ and $\sigma_{i,j}(h_{KS})$.

Otherwise, we can use the *occasional proportional feedback*, which is a variant of the original OGY chaos control method. Instead of using the unstable manifold of the attractor to compute corrections, it uses one of the dynamical variables, in a type of one-dimensional OGY method. This feedback could

be applied continuously or discretely in time; in occasional proportional feedback it is applied discretely. Occasional proportional feedback exploits the strongly dissipative nature of the flows often encountered, enabling one to control them with a one-dimensional map. The method is easy to implement, and in many cases one can stabilize high period unstable orbits by using multiple corrections per period. It is a suitable method on which to base a fuzzy logic technique for the control of chaos, since it requires no knowledge of a system model, but merely an accessible system parameter [CD98].

4.4 Evolutionary Fuzzy Recurrent Control and Computational Geometry

The development of controllers by using fuzzy logic techniques has been subject of theoretical research with many interesting successful applications produced during last years [Bab98]. The main reason is that fuzzy logic controllers (FLC) provide satisfactory performance in face of uncertainty and imprecision [Hag04], while keeping an equivalence in knowledge representation with other methods like neural networks and automata [GOT99]. An FLC represents a non linear model as the combination of a set of local linear models, where each one represents the dynamics of a complex system in a single local region [Fen02]. Each local model is specified by a fuzzy rule, which defines the local region in which the rule applies through the membership functions used in the antecedent, while the consequent defines the output of the model. Most FLCs can be classified in two categories: the Mamdani type FLC where the output is computed as a combination of fuzzy numbers, and the Takagi–Sugeno (TS) type, where the output is defined as a linear combination of the inputs. In a TS type FLC with n inputs and m outputs, a typical rule has the following form:

$$\begin{aligned}
 R_i : & \text{ if } x_1 \text{ is } A_i^1 \text{ and } \dots \text{ and } x_n \text{ is } A_i^n \\
 & \text{ then } y_1 = a_{i0}^1 + a_{ij}^1 x_j \\
 & \dots \\
 & y_m = a_{i0}^m + a_{ij}^m x_j,
 \end{aligned} \tag{4.5}$$

where x_j ($1 \leq j \leq n$) are the input variables, y_j ($1 \leq j \leq m$) are the output variables, A_i^j ($1 \leq j \leq n$) are the fuzzy membership sets and a_{ij}^k ($0 \leq j \leq n, 1 \leq k \leq m$) are the real valued parameters that define the linear approximation. The output of the complete FLC is computed by combining the outputs produced by all the rules, weighted by the degree of satisfaction of the antecedents.

Simple FLCs are usually defined by a trial and error process by using expert knowledge. However, automatic FLC generation methods are preferred for complex control systems. Most FLC structures can be mapped into feedforward neural networks, allowing the use of neural network learning algorithms

to automate the design of FLC based on numerical data as well as on expert knowledge. The combined approach provides advantages from both worlds: the low level learning and computational power of neural networks is joined together with the high level human-like thinking and reasoning of fuzzy systems [LL86]. This combination has been very successful and there is a large number of models that combines fuzzy systems with neural networks [LL86], or even with standard PID control [SE04] [HG02].

The domain of application of these systems is limited to static problems due to its feed forward network structure [LT00]. Most non linear problems in control require the processing of temporal sequences, or in other words, in these problems the output depends on the current input and previous values of inputs and/or outputs. A very interesting approach that considers small order temporal problems with fuzzy logic is proposed in [CRO04]. However, unless the number of delayed inputs and outputs is known before, it is not possible to define a feed forward model that can process temporal sequences [Jua02]. This is usually the case for most control problems, where this information is not known. However, recurrent structures can deal with this kind of problems. There is a large number of neural network models that have been proposed which are essentially feed forward structures with an extra set of units used to store previous activation values that are connected back to the inputs of other units.

By considering the amount of recurrent neural network models that have been proposed, it is not unexpected to see that most recurrent fuzzy systems are based on neural networks. For example, the model RFNN (Recurrent Fuzzy Neural Network) proposed in [LT00] defines recurrent connections in the second layer of the structure, which corresponds to the units that codifies the membership antecedent values. The model RSONFIN (Recurrent Self Organizing Neural Fuzzy Inference Network) proposed in [JL99] perform structure and parameter learning and includes an extra layer of units with recurrent connections that provides a kind of internal memory. The model DFNN (Dynamic Fuzzy Neural Network) proposed in [MT02] includes recurrent neural networks in the consequent in place of standard linear approximators like in the TS model. The TRFN (Takagi-Sugeno Type Recurrent Fuzzy Network) model proposed in [Jua02] has an extra unit with recurrent connections for each fuzzy rule, which is responsible of memorizing the temporal history of activations of the rule. Other models like [KS02] and [ZM99] follow similar approaches. In most cases, both supervised learning and non gradient based algorithms, like genetic algorithms or reinforcement learning, have been used to build or enhance the models.

Even if these recurrent models are successful in supporting learning of temporal sequences, in most cases the logic interpretation of recurrent units is not considered. In this section, following [KS04, KSR05], we present a recurrent structure for fuzzy systems based in the Fuzzy Voronoi (FV) method, which allowed the definition of recurrent fuzzy systems with a clear interpretation of recurrent units. The proposed *recurrent fuzzy voronoi (RFV) model* structure

consists of a set of rules, where the antecedent of the rules are determined by multidimensional membership functions defined in terms of Voronoi regions. The RFV model includes external and internal variables with recurrent connections that allow the processing of temporal sequences of arbitrary length. Genetic algorithms are proposed as a design tool, since they do not require derivative information, which in most control problems is unavailable or costly to get.

4.4.1 Fuzzy Recurrent Voronoi Model

Here the structure of the RFV model is presented following [KS04, KSR05]; the fuzzy reasoning strategy is explained and the details on the computation of the membership functions are introduced.

RFV Structure

A schematic diagram of the model is shown in figure 4.13, which is organized in four layers and consists of l input variables, r internal variables, m output variables and ω rules. Units in layer 1 are called input units. There are two

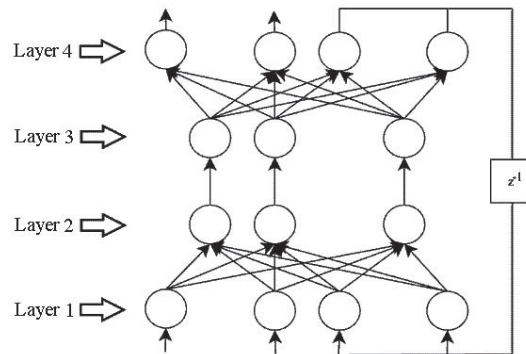


Fig. 4.13. The structure of the RFV model (adapted and modified from [KS04, KSR05].)

types of input units: external inputs and internal units that are used also as standard inputs in rule definition. Units in layer 2 are called partition units. They act as multidimensional fuzzy membership functions. Units in layer 3 are called rule units. Each fuzzy rule in the fuzzy system has a corresponding rule unit. There is a one to one correspondence with units in layer 2. Units in layer 4 are called output units. They compute the outputs as a weighted linear combination of input units, generating both the external outputs and the values of the internal units to be made available as inputs in the next time step.

The function of each type of unit is described below. In the descriptions, the external input vector of size l is denoted by x , the internal vector of size r is denoted by h , the output vector of size m is denoted by y , the complete input vector for the rules of size $l+r$ is denoted by $I = x : h$ and the complete output vector produced by the rules of size $m+r$ is denoted by $O = v : y$, where $:$ identifies the concatenation operator.

Layer 1 : No computation is performed in this layer. External input values x and previous values of internal units y are transmitted to the units in layer 2.

Layer 2 : The k th unit in this layer computes the fuzzy membership value $\mu_{S_k}(I)$ of the input vector I to the multidimensional fuzzy set S_k associated to the k th rule.

Layer 3 : Units in this layer compute a linear combination of input values based on the parameters specified by each rule, weighted by the corresponding degree of activation, as usual in TS fuzzy systems. Note that this units produce $m+r$ outputs. The output produced by the unit k that corresponds to the output variable i is:

$$O_i^k = (a_{k0}^i + a_{kj}^i I_j) \mu_k(I), \quad (4.6)$$

where the a_{kj}^i are the real valued parameters that compute the linear combination of input values associated to the rule k for the output variable i .

Layer 4 : Units in this layer compute the output vector O by computing the summation of the corresponding outputs produced by each rule. That is:

$$O_i = \sum_k O_i^k.$$

Fuzzy Reasoning for RFV Model

The RFV model performs fuzzy inference by using rules defined as follows [KS04, KSR05]:

$$\begin{aligned} R_k : & \text{ if } I \text{ is } A_k \\ & \text{ then } O_1 = a_{k0}^1 + a_{kj}^1 I_j \\ & \quad \vdots \\ & \quad O_{m+r} = a_{k0}^{m+r} + a_{kj}^{m+r} I_j \end{aligned}$$

It can be noted that there is a strong similarity with the standard definition of TS rules given in equation 4.5. Except for the fact that now a single multidimensional set is used for membership, the main difference is that the fuzzy inference involves r terms in the input vector that are output values produced in the previous time step. The fuzzy system is a dynamic fuzzy inference system with the inferred values v_i produced in time $t+1$ given by:

$$v_i(t+1) = \sum_k O_i^k(t),$$

where the computation of the value $O_i^k(t)$ (see equation 4.6) involves the input vector $x(t)$ at time t and the internal values $y(t-1)$ defined in time $t-1$.

Computation Geometry and Fuzzy Membership Functions

The domain partition strategy is based on Voronoi diagrams. A *Voronoi diagram* induces a subdivision of the space based on a set of points called *sites*. Formally [BKO98], a Voronoi diagram of a set of p points $\mathcal{P} = \{P_1, \dots, P_p\}$ is the subdivision of the plane into p cells, one for each site in $\mathcal{P}$, with the property that a point M lies in the cell corresponding to a site P_i if and only if the distance between M and P_i is smaller than the distance between P and all other P_j ($j \neq i$). A related concept is the so-called *Delaunay triangulation*

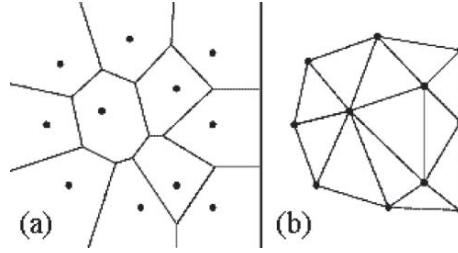


Fig. 4.14. An example of a Voronoi diagram (a) and the corresponding Delaunay triangulation (b) for a set of points in $\mathbb{R}^2$ (adapted and modified from [KS04, KSR05].)

$\mathcal{T}$, defined as the maximal planar subdivision (i.e., a subdivision such that no edge connecting two vertexes can be added to S without destroying its planarity) whose vertex set is $\mathcal{P}$ and such that the circumcircle of any triangle in $\mathcal{T}$ does not contain any point of $\mathcal{P}$ in its interior. Figure 4.14 illustrates an example of a Voronoi diagram and its corresponding Delaunay triangulation in $\mathbb{R}^2$. Note that these definitions can be straightforwardly extended to $\mathbb{R}^n$, with $n \geq 2$ – all details can be found in [BKO98].

The FV representation [KS04] considers joint fuzzy sets defined from a Voronoi diagram $\mathcal{P} = \{P_1, \dots, P_p\}$. There are as many rules as Voronoi sites. The fuzzy set S_k is defined as by its multivariate membership function μ_k that takes its maximum value 1 at site P_k , and decreases linearly to reach value 0 at the centers of all neighbor Voronoi sites. An example of such a joint fuzzy set is shown in Figure 4.15 (a) for $n = 2$.

Formally, the membership value of the input vector I to the joint fuzzy set S_k is defined by:

$$\mu_{S_k}(I) = \begin{cases} l_C(I), & x \in V_k, \\ 0, & \text{elsewhere,} \end{cases} \quad (4.7)$$

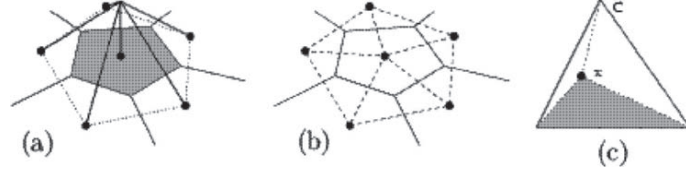


Fig. 4.15. An example of a (a) joint fuzzy set for a single Voronoi region for $n = 2$, where the membership value is represented in the z -axis, and a (b) Voronoi diagram (solid line) and its corresponding Delaunay triangulation (dotted line) for $n = 2$. The graphic (c) shows an example of the membership computation for $n = 2$. The outer triangle corresponds to the simplex defined by the Delaunay triangulation to which x belongs. The membership value corresponds to the area of the shadowed triangle. Note that the value of the area is 1 when x is equal to C and it goes down linearly to 0 on the side of the triangle opposite to C (adapted and modified from [KS04, KSR05].)

where $C = P_k$ is the Voronoi site defining S_k and the Voronoi cell V_k , and $l_C(I)$ is the barycentric coordinate of I in the simplex $T_C(I)$ of the Delaunay triangulation of $\mathcal{P}$ that has C as a vertex and to which I belongs. Figure 4.15 (b) shows an example of the Voronoi diagram and the associated Delaunay triangulation. On Figure 4.15 (c), the barycentric coordinate $l_C(I)$ corresponds to the (normalized) gray area (volume if $n > 2$) of the subsimplex formed by I and vertexes of simplex $T_C(I)$ but C . Note that a very large triangle containing all points in the domain is defined in such a way that there are no open Voronoi regions in the input domain.

4.4.2 RFV Design with Evolutionary Algorithms

Evolutionary algorithms are selected as the optimization tool for RFV controller design, since they have been very successful on problems where training data or gradient information is very difficult or costly to get, like most control problems [KS04, KSR05]. A floating point coding scheme is selected, where each individual (or chromosome) represents all free parameters of the RFV controller as a variable length vector of floating point values. An individual I with ω rules is defined as the vector:

$$Ind = R_1 : \dots : R_\omega,$$

where each sub-vector $R_i (1 \leq i \leq \omega)$ is defined as the floating point vector:

$$R_i = P_i^1 : \dots : P_i^{l+r} : a_{i0}^1 : \dots : a_{i(l+r)}^{m+r},$$

where the P_i^j are the coordinates of the site and a_{kj}^i are the real valued parameters associated to the rule R_i . The evolutionary algorithm is described in details in [SJK97, KS03]. The crossover operator is based on geometrical exchange of Voronoi sites between both parents with respect to a random

hyperplane. The mutation operator can either modify the parameters of a particular rule by some standard Gaussian mutation, or add or delete a Voronoi site or a rule.

4.4.3 Fuzzy–Genetic Properties of RFV

First of all, the RFV representation belongs to the class of approximative representations [Bab98], where each fuzzy rule defines its own fuzzy sets. It also provides continuous output, as most fuzzy systems. However, it also has a number of useful properties, that we shall now discuss in turn.

ϵ –completeness property: All RFV-based fuzzy systems defined with the RFV representation fulfills the completeness property at any required level, which establishes that any input must belong to at least one fuzzy set with a membership value not smaller than a threshold value ϵ :

$$\text{for all } x \in U : \text{ there is } A \in \{A_1, \dots, A_n\} : \text{ such that } \mu_A(x) \geq \epsilon.$$

For the RFV representation, it is clear from the definition of the membership function of equation (4.7) that this property will hold with $\epsilon = \frac{1}{2}$, as $l_C(x)$ will be above 0.5 if x lies in the Voronoi cell defined by C .

This property guarantees an adequate representation for every input point, since there is always a rule that is applied with at least a known value of membership.

No need for genetic repair algorithms: Since it is not possible to define wrong or non complete fuzzy systems, the fuzzy systems produced by applying mutation or crossover operators are always valid control systems.

Adaptive fuzzy rules: The influence on the output of a particular fuzzy rule in the RFV representation does not only depend on the rule itself, it also depends on all neighbor rules. The area of application $\mathcal{A}_k$ of a fuzzy rule R_k is defined as the union of all Delaunay regions which contain the point P_k , center of the rule R_k . Formally:

$$\mathcal{A}(R_k) = \bigcup_{P_k \in D_j} D_j :::: D_j \in D = \{D_1, \dots, D_\gamma\}.$$

where P_k is the center of the rule R_k and $D = \{D_1, \dots, D_\gamma\}$ is the Delaunay partition of the set $\mathcal{P} = \{P_1, \dots, P_p\}$.

The evolutionary algorithm evolves individuals that represent complete fuzzy systems defined by a set of fuzzy rules that are synergistically related, and not fuzzy systems defined with a set of independent fuzzy rules. The variation operators hence modify the application areas of all fuzzy rules, while still maintaining the required ϵ –completeness level.

Adaptive a‘priori rules: In most fuzzy systems, the user can incorporate a‘priori knowledge by manually defining fuzzy sets and the corresponding fuzzy rules. This process implies that some restriction on the output values and the partition of the input space is introduced in the evolutionary process, but the expected benefit is that the evolutionary process, biased toward hopefully good parts of its search space, will converge faster to better solutions.

Similarly, the RFV representation allows the definition of a‘priori rules, i.e., fixed Voronoi sites that will not be modified by evolution. But one big advantages of the RFV representation is that the expert does not need to specify the application area of such rules: thanks to the synergistic effect described above, the evolutionary process, by adding rules more or less close to the a‘priori rules will also tune its domain of application.

Recurrent rules: The rules defined in the RFV controller are standard TS type fuzzy rules, with their own inputs and outputs. The complete system is recurrent because some outputs are connected to inputs, but each rule by itself is a standard TS type fuzzy rule. This fact contributes to provide a clear interpretation of the rules and make easy to define the a‘priori rules for the RFV controller. This approach contrasts with other models like RFNN [LT00], RSONFIN [JL99], DFNN [MT02] or the TRFN [Jua02], where the rules themselves include backward connections. The recurrent connection model is similar to the NFSLS approach proposed in [MM97], except that only a single output is considered and standard fuzzy partition is performed in the input domain.

4.4.4 Application to Evolutionary Robotics

Here, following [KS04, KSR05] we apply the evolutionary approach to design RFV systems to an *evolutionary robotic problem* [NF00], where the ability to introduce a‘priori knowledge in the form of recursive rules is demonstrated. As a test base for experiments, a simulated Khepera robot [Mic04] was used for experimentation. A Khepera robot has 8 sensors that can be used to measure proximity of objects and ambient light levels, and two independent motors to control the speed and direction of the robot. The problem consists in drive the robot avoiding collisions, starting from a fixed initial position, to a target position that depends on light based signals that are set to on or off status in the trajectory. The presence of an illuminated signal (on status) indicates to the robot that it has to turn left in the next intersection, and its absence (or off status) that it has to turn right. The controller needs internal memory, since the light signal is not present in the intersection, but in a previous (and may be distant) point in the trajectory. The controller has to learn also to *forget* light signals that affected the behavior in previous intersections and have not to be considered in other point of the trajectory.

The fitness of a RFV controller is computed in a similar way as in [NF00], evaluating the controller in e different scenarios. Each scenery defines initial

and target positions, and include path intersections where light signals determine the expected trajectory of the robot. The fitness is accumulated at every step of the robot proportionally to the speed, inversely proportional to the distance to the target point and reduced when the robot travels near obstacles, in order to favor navigation without collisions. The fitness accumulation is stopped when the robot bumps an obstacle, or it reaches a maximum number of steps s . The total fitness is the average of the values obtained in the e scenarios. Formally, the fitness is defined as follows:

$$\text{fitness}(I) = \frac{1}{es} \sum_{i=1}^e \sum_{t=1}^s v(t) * (1 - a(t)) * (1 - d(t)),$$

where t is the time step, $v(t)$ is the normalized forward speed (summation of the speed of both motors), $a(t)$ is the normalized maximum activation of the sensors [NF00] (for example, $a(t) = 1$ implies a collision) and $d(t)$ is the normalized distance to the destination point (for example, $d(t) = 0$ implies that the target has been reached). This function assigns larger values to individuals that travel at the highest speed, in a trajectory that follows (when possible) a straight line, as far as possible to obstacles and minimizing the distance to the target point.

The controllers are defined with five inputs, two outputs and one internal variable. The inputs are, respectively, the average of the two left sensors, the two front sensors, the two right sensors, the two back sensors and an average of ambient light as measured by all sensors. The outputs correspond to the speed of the two motors. Note that the presence of an internal variable forces the rules to be defined with six inputs and three outputs (see Figure 4.13). In the experiments, the population size is set to 50, the probability of Voronoi crossover is set to 0.8, Voronoi mutation to 0.3, mutation for addition and removal of Voronoi sites to 0.1, selection is performed by tournament, elitism is used and the number of generations is set to 200. The performance of the individuals is measured in $e = 4$ scenarios with three intersections and all combinations of light signals, evaluated in at most $s = 500$ time steps.

The semantic of the rules is defined by considering the internal variable y_1 as a flag that indicates if a light signal was seen before. The first four rules correspond to the situation where there are no obstacles near the robot (all distance sensor values are equal to 0). The output produced in all cases for the motors is maximum forward speed (note the constant term of the approximator is 1 for both motor outputs v_1 and v_2). The value of the internal variable y_1 is set to 1 when light is present (rules 3 and 4) and to the previous value (can be 0 or 1) if no light is measured (rules 1 and 2). Rule 5 produces a turn to the right (left motor at maximum speed) if no light was detected before ($y_1 = 0$) and rule 6 a turn to the left (right motor at maximum speed) if light was detected ($y_1 = 1$). In both cases, the flag (internal variable y_1) is reset to 0. It is important to note that the a priori knowledge is defined by specifying rules that determine the expected behavior of the controller in

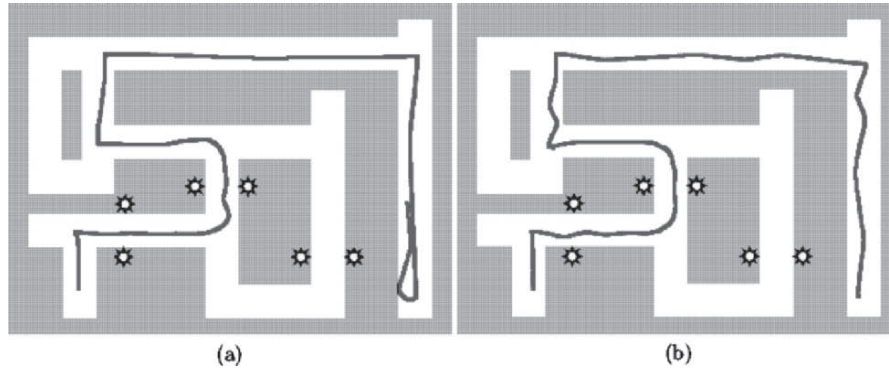


Fig. 4.16. The performance of the best controller (a) without a priori knowledge and (b) with a priori knowledge when evaluated in a scenery not used during evolution (adapted and modified from [KSR05].)

specific points in the input domain, without specifying the area of application of the rules. For more details, see [KS04, KSR05].

4.5 Fuzzy Discrete Event Systems

Recall that a *discrete event system* (DES) is a dynamical system with discrete states that are event-driven. The principal features of such systems are that they are discrete, asynchronous, and possibly nondeterministic. DES arise in a variety of contexts ranging from computer operating systems to the control of complex multi-model processes. DES theory, particularly on modeling and control, has been successfully employed in many areas such as concurrent program semantics, monitoring and control of complex systems, and communication protocols. The behavior of DES is described as the sequences of occurrences of events involved. In order to restrict the behavior of a DES to a desired range, supervisory control theory initiated by [RW87] has been intensively investigated over the last decades (see, e.g., [KG95, CL99] and references therein).

Usually, a DES is modeled by finite state automaton with events as input alphabets, and the behavior is thus the language generated by the automaton. Most of the published research on DES has been concentrated on systems that are modeled as deterministic automata or nondeterministic automata with ϵ -transitions. A common feature of these models is that their state transitions are crisp once that corresponding events occur, that is, no uncertainty arises in the state transitions. There are, however, situations in which the state transitions of some systems are always somewhat imprecise, uncertain, and vague even in a deterministic sense. A typical example given in [LY01, LY02] is a person's health status, where the change of the condition of a person's health

from a state, say ‘excellent’, to another, say ‘good’, is obviously imprecise, since it is hard to measure exactly the change. More examples will be encountered when we examine chemical reactions, mobile robots in an unstructured environment [MG02], intelligent vehicle control [Rig03], waste-water treatment [WSE00], and so on.

Uncertainty arises not only in state transitions, but also in states themselves and control strategies. The notion of a fuzzy subset of a set introduced by Zadeh in 1965 [Zad65] has been well applied to a wide range of scientific areas and has been proved to be a good method for representing uncertainty. To capture uncertainty appearing in states and state transitions of DES, Lin and Ying incorporated fuzzy set theory together with DES and thus extended crisp DES to fuzzy DES by proposing fuzzy finite automaton model in [LY01, LY02]. Under the framework of fuzzy DES, they studied generalized observability and some optimal control problems. Taking into account the uncertainty associated with control strategies, Lavrov [Lav03] introduced an enhancement of industrial control by combining discrete event supervisory control methods and fuzzy techniques.

In recent years, fuzzy model such as fuzzy automata, fuzzy Petri nets, and fuzzy neural networks, as a complement to conventional models, has become an active research topic and found successful applications in many areas [DDP04, GF02, PR03, Rig03, WSE00]. In terms of DES, *fuzzy Petri nets* have been investigated for many years [ZZ94, HLE01, RMZ03]; however, to our knowledge, few efforts, except the work [LY01, LY02], are made to consider the model and supervisory control following for fuzzy automata.

In this section, following [CY05], we present supervisory control theory for fuzzy DES by extending traditional supervisory control theory for crisp DES. The generalized supervisory control theory describes a class of DES that state transitions are imprecise and uncertain by (maxmin) fuzzy automata introduced in [San68]; the behavior of such systems is described by their generated fuzzy languages. Control is exercised by a fuzzy supervisor that disables events with certain degrees in the controlled system, the plant, so that the closed-loop system of supervisor and plant exhibits a pre-specified desired fuzzy language. We will introduce the notion of *fuzzy controllability* for fuzzy languages by generalizing crisp controllability, and present a necessary and sufficient condition for a fuzzy supervisor synthesis. When a desirable fuzzy language is not controllable, we will investigate its supremal controllable fuzzy sublanguage and infimal controllable fuzzy superlanguage, and provide their expressions and algorithms. Our results contain those relevantly known characterizations of crisp DES, as expected.

The presented work is a continuation of [LY01, LY02] (where the control problems studied are based on states and state sequences), concerned with event feedback control with fuzzy supervisor. This fuzzy automaton model also differs from that of [LY01, LY02], while the presented fuzzy supervisor is similar to the probabilistic supervisor introduced in [LW93], but resultant plant dynamics are quite distinct from each other. Fuzzy languages proposed

are similar to probabilistic languages [Paz71, GKM99, KG01], but their semantics are different: the weight of a string in a probabilistic context reflects a frequency of occurrence, while the weight in a fuzzy context describes the membership grade (namely, uncertainty) of a string. It is well known that probability theory is not capable of capturing uncertainty in all its manifestations.

4.5.1 Crisp Discrete Event Systems

Let E denote the finite set of events, and E^* denote the set of all finite strings constructed by concatenation of elements of E , including the empty string ϵ . A string $\mu \in E^*$ is a prefix of a string $\omega \in E^*$ if there exists $\nu \in E^*$ such that $\mu\nu = \omega$. In this case, we write $\mu \leq \omega$. For any $\omega \in E^*$ with length at least 1, we use $\tilde{\omega}$ to denote the maximal proper prefix of ω in what follows. In other words, $\tilde{\omega}a = \omega$ for some $a \in E$. We write $|\omega|$ for the length of a string ω . Any subset of E^* is called a language over E . The prefix closure of a language L consists of the set of strings which are prefixes of strings in L . More precisely, the prefix closure of L , denoted by $\bar{L}$, is defined by

$$\bar{L} = \{\mu \in E^* : \mu \leq \omega \text{ for some } \omega \in L\}.$$

A language L is said to be prefix closed if $L = \bar{L}$.

A DES, or plant, is usually described by a *deterministic automaton*: $G = (Q, E, \delta, q_0)$, where Q is a set of states, with the initial state q_0 , E is a set of events, and $\delta : Q \times E \rightarrow Q$ is a transition function (in general a partial function). The function δ is extended to $\delta : Q \times E^* \rightarrow Q$ in the obvious way. In a logical model of a DES, we are interested in the strings of events that the system can generate. Thus the behavior of a DES is modeled as a prefix closed language

$$L(G) = \{s \in E^* : \delta(q_0, s) \text{ is defined}\}$$

over the event set E .

The supervisory control theory partitions the event set of the system into the two disjoint sets of controllable and uncontrollable events, E_c and E_{uc} , respectively. A supervisor is a map $S : L(G) \rightarrow 2^E$ such that $S(s) \supseteq E_{uc}$ for any string $s \in L(G)$. The language generated by the controlled system is denoted by $L(S/G)$ which is defined inductively as follows:

- 1) $\epsilon \in L(S/G)$;
- 2) $[(s \in L(S/G)) \text{ and } (sa \in L(G)) \text{ and } (a \in S(s))] \Leftrightarrow [sa \in L(S/G)]$.

Let us review the controllability definition of (crisp) languages due to [RW87]: A language $K \subseteq L(G)$ is said to be *controllable* (with respect to $L(G)$ and E_{uc}) if

$$\bar{K}E_{uc} \cap L(G) \subseteq \bar{K}.$$

The key existence result for a supervisor is the following [RW87]: Given a nonempty language $K \subseteq L(G)$, there exists a supervisor S such that $L(S/G) = \bar{K}$ iff K is controllable with respect to $L(G)$ and E_{uc} .

A good property of controllable languages is that they are closed under arbitrary unions. Thus, for any language K , there exists the supremal controllable sublanguage of K [WR87], which is obtained by taking the union of all controllable sublanguages of K . Another somewhat weak property of controllable languages is that arbitrary intersections of prefix closed and controllable languages are again prefix closed and controllable. As a result, the intersection of all prefix closed and controllable suplanguages of K gives rise to the infimal prefix closed controllable suplanguages of K [LC90].

4.5.2 Fuzzy Automaton Model

Here, following [CY05] we present a fuzzy automaton model and associated fuzzy languages.

First recall some basic facts about fuzzy subsets of the universal set X . A *support* of a fuzzy set $\mathcal{A}$ is a crisp set defined as

$$\text{supp}(\mathcal{A}) = \{x : \mathcal{A}(x) > 0\}.$$

Whenever $\text{supp}(\mathcal{A})$ is a finite set, say $\text{supp}(\mathcal{A}) = \{x_1, x_2, \dots, x_n\}$, then fuzzy set $\mathcal{A}$ can be written in Zadeh's notation as follows:

$$\mathcal{A} = \frac{\mathcal{A}(x_1)}{x_1} + \frac{\mathcal{A}(x_2)}{x_2} + \dots + \frac{\mathcal{A}(x_n)}{x_n}.$$

We denote by $\mathcal{F}(X)$ the set of all fuzzy subsets of the universal set X . For any $\mathcal{A}, \mathcal{B} \in \mathcal{F}(X)$, we say that $\mathcal{A}$ is contained in $\mathcal{B}$ (or $\mathcal{B}$ contains $\mathcal{A}$), denoted by $\mathcal{A} \subseteq \mathcal{B}$, if $\mathcal{A}(x) \leq \mathcal{B}(x)$ for all $x \in X$. We say that $\mathcal{A} = \mathcal{B}$ iff $\mathcal{A} \subseteq \mathcal{B}$ and $\mathcal{B} \subseteq \mathcal{A}$. Obviously, we see that $\mathcal{A} \subseteq \mathcal{B}$ implies $\text{supp}(\mathcal{A}) \subseteq \text{supp}(\mathcal{B})$; the latter is the usual set inclusion. A fuzzy set is said to be *empty* if its membership function is identically zero on X . We use $\mathcal{O}$ to denote the empty fuzzy set.

For any family λ_i , $i \in I$, of elements of $[0, 1]$, we write $\bigvee_{i \in I} \lambda_i$ or $\bigvee \{\lambda_i : i \in I\}$ for the supremum of $\{\lambda_i : i \in I\}$, and $\bigwedge_{i \in I} \lambda_i$ or $\bigwedge \{\lambda_i : i \in I\}$ for its infimum. In particular, if I is finite, then $\bigvee_{i \in I} \lambda_i$ and $\bigwedge_{i \in I} \lambda_i$ are the greatest element and the least element in $\{\lambda_i : i \in I\}$, respectively.

Let $\mathcal{A}, \mathcal{B} \in \mathcal{F}(X)$. The *union* of $\mathcal{A}$ and $\mathcal{B}$, denoted by $\mathcal{A} \cup \mathcal{B}$, is defined by the membership function

$$(\mathcal{A} \cup \mathcal{B})(x) = \mathcal{A}(x) \vee \mathcal{B}(x), \quad \text{for all } x \in X.$$

The *intersection* of $\mathcal{A}$ and $\mathcal{B}$, denoted by $\mathcal{A} \cap \mathcal{B}$, is given by the membership function

$$(\mathcal{A} \cap \mathcal{B})(x) = \mathcal{A}(x) \wedge \mathcal{B}(x), \quad \text{for all } x \in X.$$

Now, as was mentioned in the introduction, we would like to formalize a class of DES, where the state sets are crisp and the state transitions are uncertain. We adopt a kind of fuzzy automata, which is known as maxmin automata in some mathematical literature [San68, KL79] and is somewhat

different from max-product automata used in [LY01, LY02], to model the systems. Formally, we have the following definition [CY05]: A *fuzzy automaton* is a four-tuple $G = (Q, E, \delta, q_0)$, where: Q is a crisp (finite or infinite) set of states, E is a finite set of events, $q_0 \in Q$ is the initial state, and δ is a function from $Q \times E \times Q$ to $[0, 1]$, called a *fuzzy transition function*.

For any $p, q \in Q$ and $a \in E$, we can interpret $\delta(p, a, q)$ as the possibility degree to which the automaton in state p and with the occurrence of event a may enter state q . In our graph representation of automata later, δ will be represented by arcs between states, and labels on the arcs. If p, q are states and a is an event such that $\delta(p, a, q) > 0$, then we make an arc labeled $a|\delta(p, a, q)$ from p to q .

In contrast with crisp DES, an event in fuzzy DES may take the system to more than one states with different degrees. The concept of fuzzy automata is a natural generalization of nondeterministic automata. The major difference between fuzzy automata and nondeterministic automata is: in a nondeterministic automaton, $\delta(p, a, q)$ is either 1 or 0, so the possibility degrees of existing transitions are the same, but if we work with a fuzzy automaton, they may be different.

To describe what happens when we start in any state and follow any sequence of events, we define inductively an *extended fuzzy transition function* from $Q \times E^* \times Q$ to $[0, 1]$, denoted by the same notation δ , as [CY05]:

$$\begin{aligned}\delta(p, \epsilon, q) &= \begin{cases} 1, & \text{if } q = p, \\ 0, & \text{otherwise,} \end{cases} \\ \delta(p, \omega a, q) &= \bigvee_{r \in Q} (\delta(p, \omega, r) \wedge \delta(r, a, q)),\end{aligned}$$

for all $\omega \in E^*$ and $a \in E$.

The reason we use the *maxmin rule* in the previous definition is that the state transition from p to q in a fuzzy process may be considered as water flow (gas, electricity, traffic flow, etc.) through a water supply system of which the water pipes are series-parallel inter-connected.

Further, the language $\mathcal{L}(G)$ generated by G , called *fuzzy language*, is defined as a fuzzy subset of E^* and given by

$$\mathcal{L}(G)(\omega) = \bigvee_{q \in Q} \delta(q_0, \omega, q).$$

This means that a string ω of E^* is not necessarily either ‘in the fuzzy language $\mathcal{L}(G)$ ’ or ‘not in the fuzzy language $\mathcal{L}(G)$ ’; rather ω has a membership grade $\mathcal{L}(G)(\omega)$, which measures its degree of membership in $\mathcal{L}(G)$.

Without loss of generality, all automata are considered to be *accessible automata*, that is, all states are reachable from the initial state.

If the state set of a fuzzy automaton G is empty, then necessarily $\mathcal{L}(G) = \emptyset$, that is, $\mathcal{L}(G)(\omega) = 0$ for all $\omega \in E^*$; otherwise, it is obvious by the definition that $\mathcal{L}(G)$ has the following properties:

- P1) $\mathcal{L}(G)(\epsilon) = 1$;
- P2) $\mathcal{L}(G)(\mu) \geq \mathcal{L}(G)(\mu\nu)$ for any $\mu, \nu \in E^*$.

Conversely, given a fuzzy language $\mathcal{L}$ (over the event set E) satisfying the above properties, we can construct a fuzzy automaton $G = (Q_{\mathcal{L}}, E_{\mathcal{L}}, \delta_{\mathcal{L}}, q_{\mathcal{L}})$, where

$Q_{\mathcal{L}} = \text{supp}(\mathcal{L})$, $E_{\mathcal{L}} = E$, $q_{\mathcal{L}} = \epsilon$, and

$$\delta_{\mathcal{L}}(\mu, a, \nu) = \begin{cases} \mathcal{L}(\nu), & \text{if } \nu = \mu a \\ 0, & \text{otherwise} \end{cases}$$

for all $\mu, \nu \in \text{supp}(\mathcal{L})$ and $a \in E$. There is no difficulty to verify that $\mathcal{L}(G) = \mathcal{L}$.

By a *fuzzy language* we mean the empty language $\mathcal{O}$ or a fuzzy language satisfying the above properties P1) and P2), unless otherwise specified. We use $\mathcal{FL}$ to denote the set of all fuzzy languages over E . More explicitly, $\mathcal{FL} = \{\mathcal{A} \in \mathcal{F}(E^*) : \mathcal{A} = \mathcal{O} \text{ or } \mathcal{A} \text{ satisfies P1) and P2)}\}$.

Union and intersection of fuzzy languages can be defined as usual [CY05]:

$$\begin{aligned} \left(\bigcup_{i \in I} \mathcal{L}_i\right)(\omega) &= \vee_{i \in I} \mathcal{L}_i(\omega), \text{ for all } \omega \in E^*, \text{ and} \\ \left(\bigcap_{i \in I} \mathcal{L}_i\right)(\omega) &= \wedge_{i \in I} \mathcal{L}_i(\omega), \text{ for all } \omega \in E^*. \end{aligned}$$

The concatenation $\mathcal{L}_1 \mathcal{L}_2$ of two fuzzy languages $\mathcal{L}_1$ and $\mathcal{L}_2$ is defined by

$$(\mathcal{L}_1 \mathcal{L}_2)(\omega) = \vee \{ \mathcal{L}_1(\mu) \wedge \mathcal{L}_2(\nu) : \mu, \nu \in E^* \text{ and } \mu\nu = \omega \},$$

for all $\omega \in E^*$. We sometimes write $(\mathcal{L}_1 \mathcal{L}_2)(\omega)$ as $\vee_{\mu\nu=\omega} (\mathcal{L}_1(\mu) \wedge \mathcal{L}_2(\nu))$ for convenience.

If $\mathcal{L}_1$ and $\mathcal{L}_2$ are fuzzy languages, then $\mathcal{L}_1 \cup \mathcal{L}_2$, $\mathcal{L}_1 \cap \mathcal{L}_2$, and $\mathcal{L}_1 \mathcal{L}_2$ are fuzzy languages.

If $\mathcal{L}_1 = \mathcal{O}$ or $\mathcal{L}_2 = \mathcal{O}$, then the proposition holds clearly.

Assuming now that $\mathcal{L}_1$ and $\mathcal{L}_2$ are not empty, we show that $\mathcal{L}_1 \cup \mathcal{L}_2$, $\mathcal{L}_1 \cap \mathcal{L}_2$, and $\mathcal{L}_1 \mathcal{L}_2$ satisfy the properties P1) and P2), respectively. By the definitions, it is obvious that $(\mathcal{L}_1 \cup \mathcal{L}_2)(\epsilon) = (\mathcal{L}_1 \cap \mathcal{L}_2)(\epsilon) = (\mathcal{L}_1 \mathcal{L}_2)(\epsilon) = 1$.

For any $\mu, \nu \in E^*$, we have that [CY05]

$$\begin{aligned} (\mathcal{L}_1 \cup \mathcal{L}_2)(\mu\nu) &= \mathcal{L}_1(\mu\nu) \vee \mathcal{L}_2(\mu\nu) \\ &\leq \mathcal{L}_1(\mu) \vee \mathcal{L}_2(\mu) \\ &= (\mathcal{L}_1 \cup \mathcal{L}_2)(\mu), \\ &\text{and} \\ (\mathcal{L}_1 \cap \mathcal{L}_2)(\mu\nu) &= \mathcal{L}_1(\mu\nu) \wedge \mathcal{L}_2(\mu\nu) \\ &\leq \mathcal{L}_1(\mu) \wedge \mathcal{L}_2(\mu) \\ &= (\mathcal{L}_1 \cap \mathcal{L}_2)(\mu). \end{aligned}$$

For $\mathcal{L}_1 \mathcal{L}_2$, we have that $(\mathcal{L}_1 \mathcal{L}_2)(\mu\nu) = \vee_{st=\mu\nu} (\mathcal{L}_1(s) \wedge \mathcal{L}_2(t))$ by definition. Since $|\mu\nu|$ is finite, $\vee_{st=\mu\nu} (\mathcal{L}_1(s) \wedge \mathcal{L}_2(t))$ is the maximum of $\{\mathcal{L}_1(s) \wedge \mathcal{L}_2(t) : s, t \in E^* \text{ and } st = \mu\nu\}$, which is denoted by $\mathcal{L}_1(s_0) \wedge \mathcal{L}_2(t_0)$. To prove

$(\mathcal{L}_1\mathcal{L}_2)(\mu) \geq (\mathcal{L}_1\mathcal{L}_2)(\mu\nu)$, it is sufficient to find out s' and t' that satisfy $s't' = \mu$ and $\mathcal{L}_1(s') \wedge \mathcal{L}_2(t') \geq \mathcal{L}_1(s_0) \wedge \mathcal{L}_2(t_0)$, since $(\mathcal{L}_1\mathcal{L}_2)(\mu) = \bigvee_{s't'=\mu} (\mathcal{L}_1(s') \wedge \mathcal{L}_2(t'))$. If $s_0 \leq \mu$, say $s_0s_1 = \mu$, then $s_1 \leq t_0$. Taking $s' = s_0$ and $t' = s_1$, we get that $s't' = \mu$ and $\mathcal{L}_1(s') \wedge \mathcal{L}_2(t') \geq \mathcal{L}_1(s_0) \wedge \mathcal{L}_2(t_0)$. If s_0 is not a prefix of μ , then this means that $\mu \leq s_0$. Then take $s' = \mu$ and $t' = \epsilon$. We can also get that $s't' = \mu$ and $\mathcal{L}_1(s') \wedge \mathcal{L}_2(t') \geq \mathcal{L}_1(s_0) \wedge \mathcal{L}_2(t_0)$, thus finishing the proof.

Now we have [CY05]:

- (i) $\text{supp}(\mathcal{L}_1 \cup \mathcal{L}_2) = \text{supp}(\mathcal{L}_1) \cup \text{supp}(\mathcal{L}_2);$
- (ii) $\text{supp}(\mathcal{L}_1 \cap \mathcal{L}_2) = \text{supp}(\mathcal{L}_1) \cap \text{supp}(\mathcal{L}_2);$
- (iii) $\text{supp}(\mathcal{L}_1\mathcal{L}_2) = \text{supp}(\mathcal{L}_1)\text{supp}(\mathcal{L}_2).$

In crisp DES, one can build the overall model by building models of individual components first and then composing them by product or parallel composition. Now we will extend these operations to fuzzy automata.

Given two fuzzy automata $G_i = (Q_i, E_i, \delta_i, q_{0i})$, $i = 1, 2$, we define two operations on them: the product $G_1 \times G_2$ and the parallel composition $G_1 \parallel G_2$. These operations model two forms of joint behavior of a set of fuzzy automata and we can think of them as two types of systems resulting from the interconnection of system components G_1 and G_2 .

The *product* of G_1 and G_2 is the *fuzzy automaton*

$$G_1 \times G_2 = (Q_1 \times Q_2, E_1 \cap E_2, \delta, (q_{01}, q_{02})), \quad \text{where}$$

$$\delta((p_1, q_1), a, (p_2, q_2)) = \delta_1(p_1, a, p_2) \wedge \delta_2(q_1, a, q_2)$$

for all $(p_i, q_i) \in Q_1 \times Q_2$ and $a \in E_1 \cap E_2$.

As for crisp automata, it is easily verified that $\mathcal{L}(G_1 \times G_2) = \mathcal{L}(G_1) \cap \mathcal{L}(G_2)$. This shows that the intersection of two fuzzy languages can be implemented by doing the product of their automaton representations.

The *parallel composition* of G_1 and G_2 is the fuzzy automaton [CY05]

$$G_1 \parallel G_2 = (Q_1 \times Q_2, E_1 \cup E_2, \delta, (q_{01}, q_{02})),$$

where

$$\delta((p_1, q_1), a, (p_2, q_2)) = \begin{cases} \delta_1(p_1, a, p_2) \wedge \delta_2(q_1, a, q_2), & \text{if } \delta_1(p_1, a, p_2) > 0 \\ & \text{and } \delta_2(q_1, a, q_2) > 0 \\ \delta_1(p_1, a, p_2), & \text{if } a \in E_1 \setminus E_2 \text{ and } q_1 = q_2 \\ \delta_2(q_1, a, q_2), & \text{if } a \in E_2 \setminus E_1 \text{ and } p_1 = p_2 \\ 0, & \text{otherwise.} \end{cases}$$

If $E_1 = E_2$, then the parallel composition reduces to the product, since all fuzzy transitions are forced to be synchronized.

4.5.3 Controllability of Fuzzy Languages

Here, following [CY05], we first define fuzzy supervisor and the controllability of fuzzy languages. Based on the two basic definitions, we then study what controlled behavior can be achieved when controlling a fuzzy DES with a fuzzy supervisor.

Our fuzzy DES model as described so far is simply a spontaneous generator of fuzzy event strings without a means of external control. In order to control a fuzzy DES, we postulate that certain events of the system can be disabled (i.e., prevented from occurring) with any degree when desired. This enables us to influence the evolution of the system by preventing from the occurrence degrees of some events at certain times. To model such control we partition the event set E into *controllable* and *uncontrollable* events: $E = E_c \dot{\cup} E_{uc}$, as usually done in crisp DES. However, unlike controllable events in crisp DES, a controllable event in fuzzy DES can be disabled with any degree. In other words, we allow that a controllable event in fuzzy DES can occur incompletely. But uncontrollable events can never be disabled.

Control is achieved by means of a fuzzy supervisor, which is allowed to disable any fuzzy sets of controllable events after having observed an arbitrary string $s \in \text{supp}(\mathcal{L}(G))$. Formally, a *fuzzy supervisor* for G is a map [CY05]

$$S : \text{supp}(\mathcal{L}(G)) \rightarrow \mathcal{F}(E), \quad \text{satisfying} \\ S(s)(a) = 1 \quad \text{for any } s \in \text{supp}(\mathcal{L}(G)) \text{ and } a \in E_{uc}.$$

The controlled system is denoted by S/G ; the behavior of S/G is the fuzzy language $\mathcal{L}^S$ obtained inductively as follows:

- (1) $\mathcal{L}^S(\epsilon) = 1$;
- (2) $\mathcal{L}^S(sa) = \mathcal{L}(G)(sa) \wedge S(s)(a) \wedge \mathcal{L}^S(s)$ for any $s \in E^*$ and $a \in E$.

The empty string ϵ is always in $\text{supp}(\mathcal{L}^S)$ since it is always contained in $\text{supp}(\mathcal{L}(G))$; here we have excluded the degenerate case where G is the empty automaton. Clearly, it follows from the definition of $\mathcal{L}^S$ that $\mathcal{L}^S \subseteq \mathcal{L}(G)$ and $\mathcal{L}^S \in \mathcal{FL}$.

Before giving the key definition, we have to introduce a fuzzy subset $\mathcal{E}_{uc}$ of E^* which is given by

$$\mathcal{E}_{uc}(\omega) = \begin{cases} 1, & \text{if } \omega \in E_{uc} \\ 0, & \text{if } \omega \in E^* \setminus E_{uc}. \end{cases}$$

A fuzzy language $\mathcal{K} \subseteq \mathcal{L}(G)$ is said to be *controllable* with respect to $\mathcal{L}(G)$ and E_{uc} if

$$\mathcal{K}\mathcal{E}_{uc} \cap \mathcal{L}(G) \subseteq \mathcal{K}.$$

The last expression reminds us of $\overline{K}E_{uc} \cap L(G) \subseteq \overline{K}$ in the controllability definition of crisp languages. Indeed, if all the membership grades in $\mathcal{K}$ and $\mathcal{L}(G)$ are either 0 or 1, then the two expressions are the same, since the supports of $\mathcal{K}$ is prefix closed.

It is clear that the fuzzy languages $\mathcal{O}$ and $\mathcal{L}(G)$ are always controllable with respect to $\mathcal{L}(G)$ and any E_{uc} . Unless otherwise specified, controllability will always be with respect to $\mathcal{L}(G)$ and E_{uc} , and we simply call $\mathcal{K}$ controllable if the context is clear. Sometimes, we also write $\mathcal{L}$ for $\mathcal{L}(G)$. For convenience, we will exclude the special discussions on the empty fuzzy language in the proofs of our results since $\mathcal{O}$ is trivial and the results are obvious.

A fuzzy language $\mathcal{K}$ is controllable iff $\mathcal{K}(s) \wedge \mathcal{L}(G)(sa) = \mathcal{K}(sa)$ for any $s \in E^*$ and $a \in E_{uc}$.

Suppose that $\mathcal{K}$ is controllable. Using the controllability definition and $a \in E_{uc}$, we have that $\mathcal{K}(s) \wedge \mathcal{L}(G)(sa) = (\mathcal{K}\mathcal{E}_{uc} \cap \mathcal{L}(G))(sa) \leq \mathcal{K}(sa)$. On the other hand, it is clear that $\mathcal{K}(sa) \leq \mathcal{K}(s) \wedge \mathcal{L}(G)(sa)$. This yields that $\mathcal{K}(s) \wedge \mathcal{L}(G)(sa) = \mathcal{K}(sa)$.

Conversely, assuming that $\mathcal{K}(s) \wedge \mathcal{L}(G)(sa) = \mathcal{K}(sa)$ for any $s \in E^*$ and $a \in E_{uc}$, we need prove that $\mathcal{K}\mathcal{E}_{uc} \cap \mathcal{L}(G) \subseteq \mathcal{K}$, namely $(\mathcal{K}\mathcal{E}_{uc} \cap \mathcal{L}(G))(\omega) \leq \mathcal{K}(\omega)$ for any $\omega \in E^*$. If $|\omega| = 0$, namely $\omega = \epsilon$, then it is obvious that $(\mathcal{K}\mathcal{E}_{uc} \cap \mathcal{L}(G))(\omega) = 0 \leq \mathcal{K}(\omega)$. In the case of $|\omega| > 0$, we can write ω as $\tilde{\omega}a$ for some $a \in E$. Further, if $a \in E_c$, then it is also obvious that $(\mathcal{K}\mathcal{E}_{uc} \cap \mathcal{L}(G))(\omega) = 0 \leq \mathcal{K}(\omega)$; otherwise, by assumption we have that

$$(\mathcal{K}\mathcal{E}_{uc} \cap \mathcal{L}(G))(\omega) = \mathcal{K}(\tilde{\omega}) \wedge \mathcal{L}(G)(\tilde{\omega}a) = \mathcal{K}(\tilde{\omega}a) = \mathcal{K}(\omega).$$

Overall, we get that $(\mathcal{K}\mathcal{E}_{uc} \cap \mathcal{L}(G))(\omega) \leq \mathcal{K}(\omega)$ for any $\omega \in E^*$.

The above proposition thus provides an equivalent definition of the controllability of fuzzy languages. The following theorem shows us when a desired fuzzy language can be synthesized through a fuzzy supervisor [CY05]. Let $\mathcal{K} \subseteq \mathcal{L}(G)$, where $\mathcal{K}$ is a nonempty fuzzy language. Then there exists a fuzzy supervisor S for G such that $\mathcal{L}^S = \mathcal{K}$ iff $\mathcal{K}$ is controllable. First we prove the sufficiency by constructing a desired fuzzy supervisor. For any $s \in \text{supp}(\mathcal{L}(G))$, define $S(s)$ according to

$$S(s)(a) = \begin{cases} 1, & \text{if } a \in E_{uc} \\ \mathcal{K}(sa), & \text{if } a \in E_c \end{cases}$$

for all $a \in E$. Obviously, S is a fuzzy supervisor. Then we have: $\mathcal{L}^S = \mathcal{K}$, which will be proved by induction on the length of the strings s .

- The basis step is for strings of length 0, namely, $s = \epsilon$. By assumption, $\mathcal{K}$ is a nonempty fuzzy language, so $\mathcal{K}(\epsilon) = 1$ by definition. On the other hand, we always have that $\mathcal{L}^S(\epsilon) = 1$ by definition. Thus the basis step is true.

- Assume that for all strings s with $|s| \leq n$, we have that $\mathcal{L}^S(s) = \mathcal{K}(s)$. We now prove the same for strings of the form sa , where $a \in E$. If $s \notin \text{supp}(\mathcal{L}(G))$, then $\mathcal{L}(G)(s) = 0$. Therefore $\mathcal{L}(G)(sa) = 0$ by the definition of fuzzy languages. Noting that $\mathcal{L}^S \subseteq \mathcal{L}(G)$ and $\mathcal{K} \subseteq \mathcal{L}(G)$, we get that $\mathcal{L}^S(sa) = \mathcal{K}(sa) = 0$. If $s \in \text{supp}(\mathcal{L}(G))$, then we can define $S(s)$ as above, and two cases arise.

Case 1: $a \in E_{uc}$. In this case, we get that $S(s)(a) = 1$ by definition. Thus,

$$\begin{aligned}
\mathcal{L}^S(sa) &= \mathcal{L}(G)(sa) \wedge S(s)(a) \wedge \mathcal{L}^S(s) \text{ (by definition)} \\
&= \mathcal{L}(G)(sa) \wedge \mathcal{L}^S(s) \\
&= \mathcal{L}(G)(sa) \wedge \mathcal{K}(s) \text{ (by induction hypothesis)} \\
&= \mathcal{K}(sa), \quad \text{i.e.,} \quad \mathcal{L}^S(sa) = \mathcal{K}(sa).
\end{aligned}$$

Case 2: $a \in E_c$. Using the definition of $S(s)$ and the induction hypothesis, we have the following:

$$\begin{aligned}
\mathcal{L}^S(sa) &= \mathcal{L}(G)(sa) \wedge S(s)(a) \wedge \mathcal{L}^S(s) \\
&= \mathcal{L}(G)(sa) \wedge \mathcal{K}(sa) \wedge \mathcal{L}^S(s) \\
&= \mathcal{L}(G)(sa) \wedge \mathcal{K}(sa) \wedge \mathcal{K}(s) \\
&= \mathcal{L}(G)(sa) \wedge \mathcal{K}(sa) \\
&= \mathcal{K}(sa), \quad \text{i.e.,} \quad \mathcal{L}^S(sa) = \mathcal{K}(sa).
\end{aligned}$$

The claim has been proved, thus finishing the proof of the sufficiency.

Next, to see the necessity, suppose that there is a fuzzy supervisor S for G such that $\mathcal{L}^S = \mathcal{K}$. To show the controllability of $\mathcal{K}$, by above proposition it suffices to prove that $\mathcal{L}^S(s) \wedge \mathcal{L}(G)(sa) = \mathcal{L}^S(sa)$ for any $s \in E^*$ and $a \in E_{uc}$. In fact, by definition we have that

$$\mathcal{L}^S(sa) = \mathcal{L}(G)(sa) \wedge S(s)(a) \wedge \mathcal{L}^S(s) = \mathcal{L}^S(s) \wedge \mathcal{L}(G)(sa)$$

since $S(s)(a) = 1$ for $a \in E_{uc}$. This completes the proof of the theorem.

From the above theorem, we find that such a fuzzy supervisor is not unique in general. That is, if there exists a fuzzy supervisor, say S_0 , such that $\mathcal{L}^{S_0} = \mathcal{K}$, then there are generally more than one fuzzy supervisors satisfying the requirements. In fact, more fuzzy supervisors can be constructed by

$$S(s)(a) = \begin{cases} 1, & \text{if } a \in E_{uc} \\ \mathcal{K}(sa), & \text{if } a \in E_c \text{ and} \\ & \mathcal{L}(sa) \wedge \mathcal{K}(s) > \mathcal{K}(sa) \\ x \in [\mathcal{K}(sa), 1], & \text{if } a \in E_c \text{ and} \\ & \mathcal{L}(sa) \wedge \mathcal{K}(s) = \mathcal{K}(sa). \end{cases}$$

Imitating the above proof of the sufficiency, we can easily prove that $\mathcal{L}^S = \mathcal{K}$.

For the sake of illustrating the above theorem and remark, let us consider a very simple example.

For example, let $E = \{a, b\}$ and $E_{uc} = \{b\}$. $\mathcal{L}$ and $\mathcal{K}$ are given by

$$\begin{aligned}
\mathcal{L} &= \frac{1}{\epsilon} + \frac{0.9}{a} + \frac{0.7}{aa} + \frac{0.7}{ab} + \frac{0.5}{aba}, \\
\mathcal{K} &= \frac{1}{\epsilon} + \frac{0.8}{a} + \frac{0.7}{aa} + \frac{0.7}{ab}.
\end{aligned}$$

Take a fuzzy supervisor S as follows:

$$S(\epsilon) = \frac{0.8}{a} + \frac{1}{b}, \quad S(a) = \frac{0.7}{a} + \frac{1}{b},$$

$$S(aa) = S(ab) = S(aba) = \frac{0}{a} + \frac{1}{b}.$$

Then we see that $\mathcal{L}^S = \mathcal{K}$. In fact, we can also achieve this aim by taking the following fuzzy supervisor:

$$S(\epsilon) = \frac{0.8}{a} + \frac{1}{b}, \quad S(a) = \frac{x}{a} + \frac{1}{b}, \quad S(aa) = \frac{y}{a} + \frac{1}{b},$$

$$S(ab) = \frac{0}{a} + \frac{1}{b}, \quad S(aba) = \frac{z}{a} + \frac{1}{b},$$

where $x \in [0.7, 1]$ and $y, z \in [0, 1]$.

Based on the above theorem, we can now formulate the following Supervisory Control Problem (SCP) for fuzzy DES [CY05]: Given fuzzy DES G with event set E , uncontrollable event set $E_{uc} \subseteq E$, and two fuzzy languages $\mathcal{L}_a$ and $\mathcal{L}_l$, where $\emptyset \neq \mathcal{L}_a \subseteq \mathcal{L}_l \subseteq \mathcal{L}(G)$, find a fuzzy supervisor S such that $\mathcal{L}_a \subseteq \mathcal{L}^S \subseteq \mathcal{L}_l$.

Here, $\mathcal{L}_a$ describes the minimal acceptable behavior and $\mathcal{L}_l$ describes the maximal legal behavior. SCP requires to find a fuzzy supervisor such that the behavior of controlled system is both acceptable and legal.

The following fact shows that controllable fuzzy languages are closed under arbitrary unions and intersections, respectively. If fuzzy languages $\mathcal{K}_i$, $i \in I$, are controllable, then so are $\bigcup_{i \in I} \mathcal{K}_i$ and $\bigcap_{i \in I} \mathcal{K}_i$. For simplicity, we prove the case that I is finite. There is no difficulty to generalize the proof to infinite index set. If $\mathcal{K}_1$ and $\mathcal{K}_2$ are controllable fuzzy languages, then by above proposition we know that $\mathcal{K}_1 \cup \mathcal{K}_2$ and $\mathcal{K}_1 \cap \mathcal{K}_2$ are fuzzy languages. We first show that $\mathcal{K}_1 \cup \mathcal{K}_2$ is controllable. For any $s \in E^*$ and $a \in E_{uc}$, we have that

$$\begin{aligned} & (\mathcal{K}_1 \cup \mathcal{K}_2)(s) \wedge \mathcal{L}(sa) \\ &= (\mathcal{K}_1(s) \vee \mathcal{K}_2(s)) \wedge \mathcal{L}(sa) \\ &= (\mathcal{K}_1(s) \wedge \mathcal{L}(sa)) \vee (\mathcal{K}_2(s) \wedge \mathcal{L}(sa)) \\ &= \mathcal{K}_1(sa) \vee \mathcal{K}_2(sa) \quad (\text{by controllability}) \\ &= (\mathcal{K}_1 \cup \mathcal{K}_2)(sa), \quad \text{i.e.,} \\ & (\mathcal{K}_1 \cup \mathcal{K}_2)(s) \wedge \mathcal{L}(sa) = (\mathcal{K}_1 \cup \mathcal{K}_2)(sa). \end{aligned}$$

Hence, $\mathcal{K}_1 \cup \mathcal{K}_2$ is controllable by above proposition. Similarly, one can prove that $\mathcal{K}_1 \cap \mathcal{K}_2$ is controllable. Thus the proposition follows when I is finite.

Let us turn now to discussing what can be done when the given fuzzy language $\mathcal{K}$ is uncontrollable. If $\mathcal{K}$ is not controllable, then it is natural to consider the possibility of approximating $\mathcal{K}$ by some controllable fuzzy languages. As in crisp DES, two questions arise: Is there the ‘largest’ controllable fuzzy sublanguage of $\mathcal{K}$, and is there the ‘least’ controllable fuzzy superlanguage of $\mathcal{K}$, where ‘largest’ and ‘least’ are in terms of fuzzy set inclusion. Both

answers are positive, and we address the two questions in the subsequent two sections, respectively.

Now we define the class of controllable fuzzy sublanguages of $\mathcal{K}$ and the class of controllable fuzzy superlanguages of $\mathcal{K}$ as follows:

$$\begin{aligned} C_{sub}(\mathcal{K}) &= \{\mathcal{M} \in \mathcal{FL} : \mathcal{M} \subseteq \mathcal{K} \text{ and } \mathcal{M} \text{ is controllable}\}, \quad \text{and} \\ C_{sup}(\mathcal{K}) &= \{\mathcal{M} \in \mathcal{FL} : \mathcal{K} \subseteq \mathcal{M} \subseteq \mathcal{L} \text{ and } \mathcal{M} \text{ is controllable}\}. \end{aligned}$$

Observe that $\mathcal{O} \in C_{sub}(\mathcal{K})$ and $\mathcal{L} \in C_{sup}(\mathcal{K})$, so the two classes are not empty.

Supremal Controllable Fuzzy Sublanguage

Define $\mathcal{K}^\uparrow = \bigcup_{\mathcal{M} \in C_{sub}(\mathcal{K})} \mathcal{M}$. Then the existence of the largest controllable fuzzy sublanguage of $\mathcal{K}$ follows from the following proposition [CY05]. The fuzzy language $\mathcal{K}^\uparrow$ is the largest controllable fuzzy sublanguage of $\mathcal{K}$.

Since by definition $\mathcal{M} \subseteq \mathcal{K}^\uparrow$ for any $\mathcal{M} \in C_{sub}(\mathcal{K})$, we call $\mathcal{K}^\uparrow$ the *supremal controllable fuzzy sublanguage* of $\mathcal{K}$. If $\mathcal{K}$ is controllable, then $\mathcal{K}^\uparrow = \mathcal{K}$. In the ‘worst’ case, $\mathcal{K}^\uparrow = \mathcal{O}$. We will refer to ‘ $\uparrow$ ’ as the operation of obtaining the supremal controllable fuzzy sublanguage.

Several useful properties of the operation are presented in the following proposition. Let $\mathcal{K}_1, \mathcal{K}_2 \in \mathcal{FL}$.

- 1) If $\mathcal{K}_1 \subseteq \mathcal{K}_2$, then $\mathcal{K}_1^\uparrow \subseteq \mathcal{K}_2^\uparrow$.
- 2) $(\mathcal{K}_1 \cap \mathcal{K}_2)^\uparrow = \mathcal{K}_1^\uparrow \cap \mathcal{K}_2^\uparrow$.
- 3) $\mathcal{K}_1^\uparrow \cup \mathcal{K}_2^\uparrow \subseteq (\mathcal{K}_1 \cup \mathcal{K}_2)^\uparrow$.

As a proof, 1) is clear by the definition of the operation $\uparrow$.

2) Using 1), we get that $(\mathcal{K}_1 \cap \mathcal{K}_2)^\uparrow \subseteq \mathcal{K}_1^\uparrow$ and $(\mathcal{K}_1 \cap \mathcal{K}_2)^\uparrow \subseteq \mathcal{K}_2^\uparrow$, so $(\mathcal{K}_1 \cap \mathcal{K}_2)^\uparrow \subseteq \mathcal{K}_1^\uparrow \cap \mathcal{K}_2^\uparrow$. By contradiction, suppose that $(\mathcal{K}_1 \cap \mathcal{K}_2)^\uparrow \subsetneq \mathcal{K}_1^\uparrow \cap \mathcal{K}_2^\uparrow$. Then there exists $s \in E^*$ such that $(\mathcal{K}_1 \cap \mathcal{K}_2)^\uparrow(s) < \mathcal{K}_1^\uparrow(s) \wedge \mathcal{K}_2^\uparrow(s)$, that is, $\mathcal{K}_i^\uparrow(s) > (\mathcal{K}_1 \cap \mathcal{K}_2)^\uparrow(s)$ for $i = 1, 2$. By definition, $\mathcal{K}_i^\uparrow(s) = \vee\{\mathcal{M}_i(s) : \mathcal{M}_i \in C_{sub}(\mathcal{K}_i)\}$ and $(\mathcal{K}_1 \cap \mathcal{K}_2)^\uparrow(s) = \vee\{\mathcal{M}(s) : \mathcal{M} \in C_{sub}(\mathcal{K}_1 \cap \mathcal{K}_2)\}$. Therefore there is at least one $\mathcal{M}'_i \in C_{sub}(\mathcal{K}_i)$ such that $\mathcal{M}'_i(s) > \mathcal{M}(s)$ for all $\mathcal{M} \in C_{sub}(\mathcal{K}_1 \cap \mathcal{K}_2)$. Otherwise, it contradicts with $\mathcal{K}_i^\uparrow(s) > (\mathcal{K}_1 \cap \mathcal{K}_2)^\uparrow(s)$. Hence $(\mathcal{M}'_1 \cap \mathcal{M}'_2)(s) > \mathcal{M}(s)$ for all $\mathcal{M} \in C_{sub}(\mathcal{K}_1 \cap \mathcal{K}_2)$. Observe that $\mathcal{M}'_1 \cap \mathcal{M}'_2 \in C_{sub}(\mathcal{K}_1 \cap \mathcal{K}_2)$ since $\mathcal{M}'_1 \cap \mathcal{M}'_2 \subseteq \mathcal{K}_1 \cap \mathcal{K}_2$ and $\mathcal{M}'_1 \cap \mathcal{M}'_2$ is controllable by above proposition. Taking $\mathcal{M} = \mathcal{M}'_1 \cap \mathcal{M}'_2$, then we get $(\mathcal{M}'_1 \cap \mathcal{M}'_2)(s) > (\mathcal{M}'_1 \cap \mathcal{M}'_2)(s)$, which is absurd. Thus $(\mathcal{K}_1 \cap \mathcal{K}_2)^\uparrow = \mathcal{K}_1^\uparrow \cap \mathcal{K}_2^\uparrow$.

3) By 1), we get that $\mathcal{K}_1^\uparrow \subseteq (\mathcal{K}_1 \cup \mathcal{K}_2)^\uparrow$ and $\mathcal{K}_2^\uparrow \subseteq (\mathcal{K}_1 \cup \mathcal{K}_2)^\uparrow$, so $\mathcal{K}_1^\uparrow \cup \mathcal{K}_2^\uparrow \subseteq (\mathcal{K}_1 \cup \mathcal{K}_2)^\uparrow$.

In general, $\mathcal{K}_1^\uparrow \cup \mathcal{K}_2^\uparrow = (\mathcal{K}_1 \cup \mathcal{K}_2)^\uparrow$ cannot hold. For example, let

$$E = E_{uc} = \{a, b\}, \quad \mathcal{L} = \frac{1}{\epsilon} + \frac{0.8}{a} + \frac{0.5}{b},$$

$$\mathcal{K}_1 = \frac{1}{\epsilon} + \frac{0.7}{a} + \frac{0.5}{b}, \text{ and } \mathcal{K}_2 = \frac{1}{\epsilon} + \frac{0.8}{a} + \frac{0.4}{b}.$$

Then $\mathcal{K}_1^\dagger \cup \mathcal{K}_2^\dagger = \mathcal{O} \cup \mathcal{O} = \mathcal{O}$, whereas $(\mathcal{K}_1 \cup \mathcal{K}_2)^\dagger = \mathcal{L}^\dagger = \mathcal{L}$.

Next, we wish to give an explicit expression of the membership function associated with $\mathcal{K}^\dagger$. For this purpose, let us first define inductively an auxiliary fuzzy subset $\hat{\mathcal{K}}$ of E^* :

$$\hat{\mathcal{K}}(\epsilon) = \begin{cases} 0, & \text{if } \mathcal{K}(\omega) < \mathcal{L}(\omega) \text{ for some } \omega \in E_{uc}^* \\ 1, & \text{otherwise;} \end{cases}$$

$$\hat{\mathcal{K}}(sa) = \begin{cases} \hat{\mathcal{K}}(s) \wedge \mathcal{K}(sa), & \text{if } a \in E_{uc} \\ \hat{\mathcal{K}}(s) \wedge \mathcal{K}(sa) \wedge \lambda(sa), & \text{if } a \in E_c \end{cases}$$

for all $s \in E^*$ and $a \in E$, where we use $\lambda(sa)$ to denote $\bigwedge \{\mathcal{K}(t) : t \in saE_{uc}^* \text{ and } \mathcal{K}(t) < \mathcal{L}(t) \wedge \mathcal{K}(\tilde{t})\}$ and use the abbreviation saE_{uc}^* for $\{s\}\{a\}E_{uc}^*$. Recall that for any $s \in E^*$ with $|s| \geq 1$, we use $\tilde{s}$ to denote the maximal proper prefix of s . If $\{\mathcal{K}(t) : t \in saE_{uc}^* \text{ and } \mathcal{K}(t) < \mathcal{L}(t) \wedge \mathcal{K}(\tilde{t})\} = \emptyset$, then we set $\lambda(sa) = 1$.

Our aim now is to show that $\mathcal{K}^\dagger = \hat{\mathcal{K}}$. To this end, it is convenient to have the following three lemmas [CY05].

1. $\hat{\mathcal{K}} \in \mathcal{FL}$. By the above definition, it is clear that $\hat{\mathcal{K}} \in \mathcal{F}(E^*)$. If $\hat{\mathcal{K}}(\epsilon) = 0$, then by induction on the length of strings we can easily get that $\hat{\mathcal{K}}(s) = 0$ for any $s \in E^*$. Thus $\hat{\mathcal{K}}$ is the empty fuzzy language when $\hat{\mathcal{K}}(\epsilon) = 0$. In the case of $\hat{\mathcal{K}}(\epsilon) = 1$, note that by definition we always have $\hat{\mathcal{K}}(sa) \leq \hat{\mathcal{K}}(s)$. This implies that $\hat{\mathcal{K}}$ satisfies the property P2). Hence we also have that $\hat{\mathcal{K}} \in \mathcal{FL}$ when $\hat{\mathcal{K}}(\epsilon) = 1$, finishing the proof.

2. $\hat{\mathcal{K}}(s) \wedge \mathcal{K}(sa) = \hat{\mathcal{K}}(s) \wedge \mathcal{L}(sa)$ for any $s \in E^*$ and $a \in E_{uc}$.

If $\hat{\mathcal{K}}(\epsilon) = 0$, then we have that $\hat{\mathcal{K}} = \mathcal{O}$ from the above lemma. So $\hat{\mathcal{K}}(s) \wedge \mathcal{K}(sa) = \hat{\mathcal{K}}(s) \wedge \mathcal{L}(sa) = 0$.

Let us consider the other case $\hat{\mathcal{K}}(\epsilon) = 1$. Two subcases need to be discussed:

Subcase 1: $s \in E_{uc}^*$. Since $\hat{\mathcal{K}}(\epsilon) = 1$, we know by definition that $\mathcal{K}(\omega) = \mathcal{L}(\omega)$ for all $\omega \in E_{uc}^*$. Thus $\mathcal{K}(sa) = \mathcal{L}(sa)$ because of $a \in E_{uc}$. Consequently, $\hat{\mathcal{K}}(s) \wedge \mathcal{K}(sa) = \hat{\mathcal{K}}(s) \wedge \mathcal{L}(sa)$.

Subcase 2: $s \notin E_{uc}^*$. In this subcase, s can be written uniquely as $s = \mu c \nu$ for some $\mu \in E^*$, $c \in E_c$, and $\nu \in E_{uc}^*$.

If $|\nu| = 0$, namely, $s = \mu c$, then by definition we have the following:

$$\begin{aligned} \hat{\mathcal{K}}(s) \wedge \mathcal{K}(sa) &= \hat{\mathcal{K}}(\mu c) \wedge \mathcal{K}(\mu c a) \\ &= \hat{\mathcal{K}}(\mu) \wedge \mathcal{K}(\mu c) \wedge \lambda(\mu c) \wedge \mathcal{K}(\mu c a) =: (1) \end{aligned}$$

$$\begin{aligned} \hat{\mathcal{K}}(s) \wedge \mathcal{L}(sa) &= \hat{\mathcal{K}}(\mu c) \wedge \mathcal{L}(\mu c a) \\ &= \hat{\mathcal{K}}(\mu) \wedge \mathcal{K}(\mu c) \wedge \lambda(\mu c) \wedge \mathcal{L}(\mu c a) =: (2). \end{aligned}$$

Clearly, $\mathcal{K}(\mu c a) \leq \mathcal{L}(\mu c a) \wedge \mathcal{K}(\mu c)$. When $\mathcal{K}(\mu c a) = \mathcal{L}(\mu c a) \wedge \mathcal{K}(\mu c)$, we get that

$$(1) = \hat{\mathcal{K}}(\mu) \wedge \mathcal{K}(\mu c) \wedge \lambda(\mu c) \wedge \mathcal{L}(\mu c a) = (2).$$

If $\mathcal{K}(\mu c a) < \mathcal{L}(\mu c a) \wedge \mathcal{K}(\mu c)$, then

$$\begin{aligned} \lambda(\mu c) &= \wedge \{ \mathcal{K}(t) : t \in \mu c E_{uc}^* \text{ and } \mathcal{K}(t) < \mathcal{L}(t) \wedge \mathcal{K}(\tilde{t}) \} \\ &\leq \mathcal{K}(\mu c a) \\ &\leq \mathcal{L}(\mu c a). \end{aligned}$$

Hence,

$$(1) = \hat{\mathcal{K}}(\mu) \wedge \mathcal{K}(\mu c) \wedge \lambda(\mu c) = (2).$$

If $|\nu| > 0$, say $|\nu| = n$, then we have that $\nu = \nu_1 \nu_2 \cdots \nu_n$ for some $\nu_i \in E_{uc}$. Thus by definition we get that

$$\hat{\mathcal{K}}(s) = \hat{\mathcal{K}}(\mu c \nu_1 \nu_2 \cdots \nu_n) = \hat{\mathcal{K}}(\mu c) \wedge \mathcal{K}(s).$$

So we get that

$$\begin{aligned} \hat{\mathcal{K}}(s) \wedge \mathcal{K}(s a) &= \hat{\mathcal{K}}(\mu c) \wedge \mathcal{K}(s) \wedge \mathcal{K}(s a) \\ &= \hat{\mathcal{K}}(\mu) \wedge \mathcal{K}(\mu c) \wedge \lambda(\mu c) \wedge \mathcal{K}(s) \wedge \mathcal{K}(s a) \\ &= \hat{\mathcal{K}}(\mu) \wedge \lambda(\mu c) \wedge \mathcal{K}(s) \wedge \mathcal{K}(s a) =: (3) \end{aligned}$$

and

$$\begin{aligned} \hat{\mathcal{K}}(s) \wedge \mathcal{L}(s a) &= \hat{\mathcal{K}}(\mu c) \wedge \mathcal{K}(s) \wedge \mathcal{L}(s a) \\ &= \hat{\mathcal{K}}(\mu) \wedge \mathcal{K}(\mu c) \wedge \lambda(\mu c) \wedge \mathcal{K}(s) \wedge \mathcal{L}(s a) \\ &= \hat{\mathcal{K}}(\mu) \wedge \lambda(\mu c) \wedge \mathcal{K}(s) \wedge \mathcal{L}(s a) =: (4). \end{aligned}$$

If $\mathcal{K}(s a) = \mathcal{L}(s a) \wedge \mathcal{K}(s)$, then we see that

$$(3) = \hat{\mathcal{K}}(\mu) \wedge \lambda(\mu c) \wedge \mathcal{K}(s) \wedge \mathcal{L}(s a) = (4).$$

If $\mathcal{K}(s a) < \mathcal{L}(s a) \wedge \mathcal{K}(s)$, then

$$\begin{aligned} \lambda(\mu c) &= \wedge \{ \mathcal{K}(t) : t \in \mu c E_{uc}^* \text{ and } \mathcal{K}(t) < \mathcal{L}(t) \wedge \mathcal{K}(\tilde{t}) \} \\ &\leq \mathcal{K}(\mu c \nu_1 \nu_2 \cdots \nu_n a) \\ &= \mathcal{K}(s a) \\ &\leq \mathcal{L}(s a). \end{aligned}$$

This gives rise to

$$(3) = \hat{\mathcal{K}}(\mu) \wedge \lambda(\mu c) \wedge \mathcal{K}(s) = (4).$$

Overall, we have that $\hat{\mathcal{K}}(s) \wedge \mathcal{K}(s a) = \hat{\mathcal{K}}(s) \wedge \mathcal{L}(s a)$ for any $s \in E^*$ and $a \in E_{uc}$, and the proof is completed.

3. Suppose that $\mathcal{M} \subseteq \mathcal{L}$ is controllable and $\mathcal{M} \neq \mathcal{O}$. Then $\mathcal{M}(\mu) = \mathcal{L}(\mu)$ for any $\mu \in E_{uc}^*$.

It is clear that $\mathcal{M}(\epsilon) = \mathcal{L}(\epsilon) = 1$ since $\mathcal{M} \neq \mathcal{O}$. Assume that $\mathcal{M}(\mu) = \mathcal{L}(\mu)$ for all strings $\mu \in E_{uc}^*$ with $|\mu| \leq n$. For any $a \in E_{uc}$, we have that

$$\begin{aligned}\mathcal{M}(\mu a) &= \mathcal{M}(\mu) \wedge \mathcal{L}(\mu a) \text{ (by the controllability of } \mathcal{M}) \\ &= \mathcal{L}(\mu) \wedge \mathcal{L}(\mu a) \text{ (by induction hypothesis)} \\ &= \mathcal{L}(\mu a), \quad \text{i.e.,} \quad \mathcal{M}(\mu a) = \mathcal{L}(\mu a).\end{aligned}$$

Based upon the above three lemmas, we now verify that [CY05]: $\mathcal{K}^\dagger = \hat{\mathcal{K}}$. Let us first show that $\hat{\mathcal{K}} \subseteq \mathcal{K}^\dagger$. It suffices to verify that

$$\hat{\mathcal{K}} \in C_{sub}(\mathcal{K}) = \{\mathcal{M} \in \mathcal{FL} : \mathcal{M} \subseteq \mathcal{K} \text{ and } \mathcal{M} \text{ is controllable}\}.$$

We see that $\hat{\mathcal{K}}$ is a fuzzy language; moreover, $\hat{\mathcal{K}} \subseteq \mathcal{K}$ by the definition of $\hat{\mathcal{K}}$. It remains only to show that $\hat{\mathcal{K}}$ is controllable. For any $s \in E^*$ and $a \in E_{uc}$, we see that $\hat{\mathcal{K}}(s) \wedge \mathcal{L}(sa) = \hat{\mathcal{K}}(s) \wedge \mathcal{K}(sa) = \hat{\mathcal{K}}(sa)$. Therefore, it follows that $\hat{\mathcal{K}}$ is controllable.

For the reverse inclusion, take any $\mathcal{M} \in C_{sub}(\mathcal{K})$. It is enough to prove that $\mathcal{M} \subseteq \hat{\mathcal{K}}$; we prove it by showing that $\mathcal{M}(s) \leq \hat{\mathcal{K}}(s)$ for any $s \in E^*$. Use induction on the length of s .

In the basis step, namely, $s = \epsilon$, if $\mathcal{M}(\epsilon) = 0$ or $\hat{\mathcal{K}}(\epsilon) = 1$, then it is obvious that $\mathcal{M}(\epsilon) \leq \hat{\mathcal{K}}(\epsilon)$. Suppose now that $\mathcal{M}(\epsilon) = 1$ and $\hat{\mathcal{K}}(\epsilon) = 0$. By $\hat{\mathcal{K}}(\epsilon) = 0$, we know that there exists $\mu \in E_{uc}^*$ such that $\mathcal{K}(\mu) < \mathcal{L}(\mu)$. So $\mathcal{M}(\mu) \leq \mathcal{K}(\mu) < \mathcal{L}(\mu)$, which contradicts with above lemma. Hence we have that $\mathcal{M}(s) \leq \hat{\mathcal{K}}(s)$ in the basis step.

The induction hypothesis is that $\mathcal{M}(s) \leq \hat{\mathcal{K}}(s)$ for all strings s satisfying $|s| \leq n$. We now prove the same for strings of the form sa . By contradiction, let us suppose that $\mathcal{M}(sa) > \hat{\mathcal{K}}(sa)$ for some $a \in E$. If $a \in E_{uc}$, then by definition and the induction hypothesis we get that $\hat{\mathcal{K}}(sa) = \hat{\mathcal{K}}(s) \wedge \mathcal{K}(sa) \geq \mathcal{M}(s) \wedge \mathcal{M}(sa) = \mathcal{M}(sa)$. This contradicts with the assumption that $\mathcal{M}(sa) > \hat{\mathcal{K}}(sa)$. If $a \in E_c$, then $\hat{\mathcal{K}}(sa) = \hat{\mathcal{K}}(s) \wedge \mathcal{K}(sa) \wedge \lambda(sa)$ by definition. Since $\hat{\mathcal{K}}(s) \geq \mathcal{M}(s) \geq \mathcal{M}(sa)$, $\mathcal{K}(sa) \geq \mathcal{M}(sa)$, and $\mathcal{M}(sa) > \hat{\mathcal{K}}(sa)$, we see that $\hat{\mathcal{K}}(sa) = \lambda(sa)$. That is,

$$\hat{\mathcal{K}}(sa) = \wedge \{\mathcal{K}(t) : t \in saE_{uc}^* \text{ and } \mathcal{K}(t) < \mathcal{L}(t) \wedge \mathcal{K}(\tilde{t})\}.$$

Since $\mathcal{M}(sa) > \hat{\mathcal{K}}(sa)$, there by the definition of infimum exists $t_0 \in saE_{uc}^*$ satisfying $\mathcal{K}(t_0) < \mathcal{L}(t_0) \wedge \mathcal{K}(\tilde{t}_0)$ such that $\mathcal{K}(t_0) < \mathcal{M}(sa)$. Obviously, $t_0 \neq sa$; otherwise, it contradicts with $\mathcal{M} \subseteq \mathcal{K}$. Thus we can write $t_0 = sa\nu_1\nu_2 \cdots \nu_m$ for some $\nu_i \in E_{uc}$, where $m \geq 1$. By the controllability of $\mathcal{M}$, we see that

$$\mathcal{M}(t_0) = \mathcal{M}(\tilde{t}_0) \wedge \mathcal{L}(t_0) = \mathcal{M}(\tilde{t}_0) \wedge \mathcal{K}(\tilde{t}_0) \wedge \mathcal{L}(t_0).$$

Using repeatedly the above proposition, we can get that $\mathcal{M}(\tilde{t}_0) = \mathcal{M}(sa) \wedge \mathcal{L}(\tilde{t}_0)$. Thus

$$\mathcal{M}(t_0) = \mathcal{M}(sa) \wedge \mathcal{L}(\tilde{t}_0) \wedge \mathcal{K}(\tilde{t}_0) \wedge \mathcal{L}(t_0) = \mathcal{M}(sa) \wedge \mathcal{K}(\tilde{t}_0) \wedge \mathcal{L}(t_0).$$

Noting that $\mathcal{M}(sa) > \mathcal{K}(t_0)$ and $\mathcal{K}(\tilde{t}_0) \wedge \mathcal{L}(t_0) > \mathcal{K}(t_0)$ by the previous arguments, we see that $\mathcal{M}(t_0) > \mathcal{K}(t_0)$, which contradicts with $\mathcal{M} \subseteq \mathcal{K}$ again. The proof of the induction step is finished, and the proof of the theorem is also completed.

According to the above theorem, we get an explicit expression of the membership function associated with $\mathcal{K}^\dagger$, as promised; however, it seems to be difficult to give a compact formula for $\mathcal{K}^\dagger$ when $\mathcal{K}$ is not controllable. Here we provide an upper bound for $\text{supp}(\mathcal{K}^\dagger)$. Before doing this, we have to define the quotient of (crisp) languages. Let $L_1, L_2 \subseteq E^*$. The quotient operation L_1/L_2 is defined as follows:

$$L_1/L_2 = \{s \in E^* : st \in L_1 \text{ for some } t \in L_2\}.$$

Now we have [CY05]:

$$\text{supp}(\mathcal{K}^\dagger) \subseteq \text{supp}(\mathcal{K}) \setminus [(\text{supp}(\mathcal{L}) \setminus \text{supp}(\mathcal{K})) / E_{uc}^*] E^*.$$

If $\text{supp}(\mathcal{K}^\dagger) = \emptyset$, namely, $\mathcal{K}^\dagger = \mathcal{O}$, then the proposition evidently holds; otherwise, take any $s \in \text{supp}(\mathcal{K}^\dagger)$. By contradiction, assume that

$$s \notin \text{supp}(\mathcal{K}) \setminus [(\text{supp}(\mathcal{L}) \setminus \text{supp}(\mathcal{K})) / E_{uc}^*] E^*.$$

This means that $s \in [(\text{supp}(\mathcal{L}) \setminus \text{supp}(\mathcal{K})) / E_{uc}^*] E^*$ since $s \in \text{supp}(\mathcal{K}^\dagger) \subseteq \text{supp}(\mathcal{K})$. Thus there exist $s_1, s_2 \in E^*$, and $\mu \in E_{uc}^*$ such that $s = s_1 s_2$ and $s_1 \mu \in \text{supp}(\mathcal{L})$, but $s_1 \mu \notin \text{supp}(\mathcal{K})$. Obviously, $\mu \neq \epsilon$. Otherwise it contradicts with the fact that $\text{supp}(\mathcal{K})$ is prefix closed. So we may suppose that $\mu = \mu_1 \mu_2 \cdots \mu_n$ for some $\mu_i \in E_{uc}$, where $n \geq 1$. Further, using recursively above proposition, we have that $\mathcal{K}^\dagger(s_1 \mu) = \mathcal{K}^\dagger(s_1) \wedge \mathcal{L}(s_1 \mu)$. Noting that $s_1 \leq s \in \text{supp}(\mathcal{K}^\dagger)$ and $s_1 \mu \in \text{supp}(\mathcal{L})$, we see that $\mathcal{K}^\dagger(s_1 \mu) > 0$, which contradicts with $s_1 \mu \notin \text{supp}(\mathcal{K})$. As s was arbitrary, the proposition has been proved.

Note that the upper bound for $\text{supp}(\mathcal{K}^\dagger)$ is optimal in the sense that it can be achieved by some fuzzy languages. However, the reverse inclusion in above proposition is not true in general, as shown by the following example. Let $E = \{a, b, c\}$ and $E_{uc} = \{b\}$. $\mathcal{L}$ and $\mathcal{K}$ are given by

$$\begin{aligned} \mathcal{L}(\epsilon) &= \mathcal{L}(a) = 1, \quad \mathcal{L}(c) = 0.5, \quad \mathcal{L}(ab^n) = 1/n \quad \text{for any } n \geq 1; \\ \mathcal{K}(\epsilon) &= \mathcal{K}(a) = 1, \quad \mathcal{K}(ab^n) = 1/(n+1) \quad \text{for any } n \geq 1. \end{aligned}$$

One can use the definition of $\hat{\mathcal{K}}$ to get that $\mathcal{K}^\dagger = 1/\epsilon$, so $\text{supp}(\mathcal{K}^\dagger) = \{\epsilon\}$.

We next provide an algorithm to compute $\mathcal{K}^\dagger$. For simplicity, we assume that $\mathcal{K}$ and $\mathcal{L}$ are deterministic regular for the moment, that is, $\mathcal{K}$ and $\mathcal{L}$ are generated by deterministic finite state fuzzy automata, respectively.

Let $G = (Q, E, \delta, q_0)$ and $H = (P, E, \delta_H, p_0)$ be deterministic finite state fuzzy automata that generate $\mathcal{L}$ and $\mathcal{K}$, respectively, i.e., $\mathcal{L}(G) = \mathcal{L}$ and $\mathcal{L}(H) = \mathcal{K}$, where it is assumed that $\mathcal{K} \subseteq \mathcal{L}(G)$.

Algorithm for $\mathcal{K}^\dagger$:

Step 1): Let $H_0 = (P_0, E, \delta_{H_0}, (p_0, q_0)) = H \times G$. Thus $\mathcal{L}(H_0) = \mathcal{K}$. States of H_0 will be denoted by pairs (p, q) . Let $i = 0$.

Step 2): Compare H_i with G . For each $(p, q) \in P_i$, if $\delta_{H_i}((p, q), a, (p', q')) = 0$ for some $(p', q') \in P \times Q$ and $a \in E_{uc}$, but $\delta(q, a, q') > 0$, then remove (p, q) and all its associated transitions from H_i . Take the accessible operation to give rise to H_{i+1} .

Step 3): If no states were removed from H_{i+1} in Step 2), go to Step 4). Else, set $i \leftarrow i + 1$ and go to Step 2).

Step 4): If H_{i+1} is the empty automaton, then we set $H' = H_{i+1}$ and STOP.

If there is a path ω consisting of uncontrollable events and beginning with the initial state (p_0, q_0) of H_{i+1} such that $\delta_{H_{i+1}}((p_0, q_0), \omega, (p', q')) < \delta(q_0, \omega, q')$, then we let H' be the empty automaton and STOP, too.

Otherwise, we reassign membership grades of controllable events appearing in the transitions of H_{i+1} . If $\delta_{H_{i+1}}((p_0, q_0), sa, (p', q')) > 0$, where $s \in E^*$ and a is a controllable event and, say, connects (p_s, q_s) and (p', q') , then set $\delta'_{H_{i+1}}((p_s, q_s), a, (p', q')) = \delta_{H_{i+1}}((p_s, q_s), a, (p', q')) \wedge \theta(sa)$. Here $\theta(sa) = \wedge \{ \delta_{H_{i+1}}((p', q'), \mu, (p_\mu, q_\mu)) : 0 < \delta_{H_{i+1}}((p', q'), \mu, (p_\mu, q_\mu)) < \delta(q', \mu, q_\mu) \text{ and } \mu \in E_{uc}^* \}$. Note that by the definition of extended fuzzy transition function, the set of distinct membership grades is finite. Thereby, the process is guaranteed to converge in finite steps. Repeat for all controllable events appearing in the transitions of H_{i+1} , and call the resulting automaton H' .

Step 5): $\mathcal{L}(H') = \mathcal{K}^\dagger$.

The following theorem verifies the correctness of the above algorithm. Let $\mathcal{L}, \mathcal{K}, G, H_i, H'$ be as in Algorithm for $\mathcal{K}^\dagger$. Then $\mathcal{L}(H') = \mathcal{K}^\dagger$. By the above theorem, it suffices to prove that $\mathcal{L}(H') = \hat{\mathcal{K}}$, i.e., $\mathcal{L}(H')(s) = \hat{\mathcal{K}}(s)$ for all $s \in E^*$ [CY05]. We prove it by induction on the length of s .

For $s = \epsilon$, we know by definition that $\hat{\mathcal{K}}(\epsilon) = 0$ iff there exists $\omega \in E_{uc}^*$ such that $\mathcal{K}(\omega) < \mathcal{L}(\omega)$. If one of these ω such that $\mathcal{L}(H_i)(\omega) = 0$, then Steps 2) and 3) force that H' is the empty automaton; otherwise, these ω survive in H_{i+1} , then by Step 4) we have that H' is also the empty automaton. Hence $\mathcal{L}(H')(\epsilon) = 0$. Conversely, if $\mathcal{L}(H')(\epsilon) = 0$, then it follows immediately that H' is the empty automaton. By the above algorithm, we know that there must be $\omega \in E_{uc}^*$ such that $\mathcal{K}(\omega) < \mathcal{L}(\omega)$. Consequently, $\hat{\mathcal{K}}(\epsilon) = 0$, which establishes the basis step.

For the induction step, suppose that $\mathcal{L}(H')(s) = \hat{\mathcal{K}}(s)$ for all $s \in E^*$ with $|s| \leq n$. Let us consider sa . Note that fuzzy automata under consideration are deterministic finite. Thus if $a \in E_{uc}$, then by construction and induction hypothesis we get that

$$\begin{aligned}
\mathcal{L}(H')(sa) &= \mathcal{L}(H')(s) \wedge \delta_{H'}((p_s, q_s), a, (p', q')) \\
&= \hat{\mathcal{K}}(s) \wedge \delta_{H'}((p_s, q_s), a, (p', q')) \\
&= \hat{\mathcal{K}}(s) \wedge \delta_{H_0}((p_s, q_s), a, (p', q')) \\
&= \hat{\mathcal{K}}(s) \wedge \mathcal{K}(s) \wedge \delta_{H_0}((p_s, q_s), a, (p', q')) \\
&= \hat{\mathcal{K}}(s) \wedge \mathcal{K}(sa) \\
&= \hat{\mathcal{K}}(sa), \quad \text{i.e.,} \quad \mathcal{L}(H')(sa) = \hat{\mathcal{K}}(sa).
\end{aligned}$$

If $a \in E_c$, then by definitions we have that

$$\begin{aligned}
\mathcal{L}(H')(sa) &= \mathcal{L}(H')(s) \wedge \delta_{H'}((p_s, q_s), a, (p', q')) \\
&= \hat{\mathcal{K}}(s) \wedge \delta_{H_{i+1}}((p_s, q_s), a, (p', q')) \wedge \theta(sa) \\
&= \hat{\mathcal{K}}(s) \wedge \delta_{H_0}((p_s, q_s), a, (p', q')) \wedge \theta(sa) \\
&= \hat{\mathcal{K}}(s) \wedge \mathcal{K}(s) \wedge \delta_{H_0}((p_s, q_s), a, (p', q')) \\
&= \hat{\mathcal{K}}(s) \wedge \mathcal{K}(sa) \wedge \theta(sa),
\end{aligned}$$

and $\hat{\mathcal{K}}(sa) = \hat{\mathcal{K}}(s) \wedge \mathcal{K}(sa) \wedge \lambda(sa)$. By construction and the definitions of $\theta(sa)$ and $\lambda(sa)$, it is easy to show that $\mathcal{K}(sa) \wedge \theta(sa) = \mathcal{K}(sa) \wedge \lambda(sa)$. Thus we also get that $\mathcal{L}(H')(sa) = \hat{\mathcal{K}}(sa)$ when $a \in E_c$.

Infimal Controllable Fuzzy Superlanguage

In parallel with the last subsection, we now study the ‘least’ controllable fuzzy superlanguage of $\mathcal{K}$ mentioned earlier and its calculation when $\mathcal{K}$ is not controllable.

Let us define $\mathcal{K}^\downarrow = \bigcap_{\mathcal{M} \in C_{sup}(\mathcal{K})} \mathcal{M}$. Similar to above proposition, we have

the following result which shows us the existence of the least controllable fuzzy superlanguage of $\mathcal{K}$ [CY05]: The fuzzy language $\mathcal{K}^\downarrow$ is the least controllable fuzzy superlanguage of $\mathcal{K}$.

Since by definition $\mathcal{K}^\downarrow \subseteq \mathcal{M}$ for any $\mathcal{M} \in C_{sup}(\mathcal{K})$, we call $\mathcal{K}^\downarrow$ the *infimal controllable fuzzy superlanguage* of $\mathcal{K}$. If $\mathcal{K}$ is controllable, then $\mathcal{K}^\downarrow = \mathcal{K}$. In the ‘worst’ case, $\mathcal{K}^\downarrow = \mathcal{L}$.

Let us proceed to find more characterizations of $\mathcal{K}^\downarrow$.

In order to give a compact formula for $\mathcal{K}^\downarrow$, we introduce a new fuzzy subset $\mathcal{E}_{uc}^*$ of E^* , which is given by

$$\mathcal{E}_{uc}^*(\omega) = \begin{cases} 1, & \text{if } \omega \in E_{uc}^* \\ 0, & \text{otherwise.} \end{cases}$$

We make the following observation on $\mathcal{E}_{uc}^*$: $(\mathcal{K}\mathcal{E}_{uc}^*)(sa) = (\mathcal{K}\mathcal{E}_{uc}^*)(s)$ for any $s \in E^*$ and $a \in E_{uc}$.

By definition, we know that $(\mathcal{K}\mathcal{E}_{uc}^*)(s) = \bigvee_{s_1 s_2 = s} (\mathcal{K}(s_1) \wedge \mathcal{E}_{uc}^*(s_2))$ and $(\mathcal{K}\mathcal{E}_{uc}^*)(sa) = \bigvee_{t_1 t_2 = sa} (\mathcal{K}(t_1) \wedge \mathcal{E}_{uc}^*(t_2))$. Since $|s|$ is finite, we can assume that $\bigvee_{s_1 s_2 = s} (\mathcal{K}(s_1) \wedge \mathcal{E}_{uc}^*(s_2)) = \mathcal{K}(s'_1) \wedge \mathcal{E}_{uc}^*(s'_2)$ for some s'_1 and s'_2 satisfying

$s'_1 s'_2 = s$. Setting $t'_1 = s'_1$ and $t'_2 = s'_2 a$, we have that $t'_1 t'_2 = sa$ and $\mathcal{K}(t'_1) \wedge \mathcal{E}_{uc}^*(t'_2) = \mathcal{K}(s'_1) \wedge \mathcal{E}_{uc}^*(s'_2)$. As a result, $(\mathcal{K}\mathcal{E}_{uc}^*)(sa) = \bigvee_{t_1 t_2 = sa} (\mathcal{K}(t_1) \wedge \mathcal{E}_{uc}^*(t_2)) \geq \mathcal{K}(t'_1) \wedge \mathcal{E}_{uc}^*(t'_2) = \mathcal{K}(s'_1) \wedge \mathcal{E}_{uc}^*(s'_2) = (\mathcal{K}\mathcal{E}_{uc}^*)(s)$, that is, $(\mathcal{K}\mathcal{E}_{uc}^*)(sa) \geq (\mathcal{K}\mathcal{E}_{uc}^*)(s)$. On the other hand, we observe that $\mathcal{K}\mathcal{E}_{uc}^* \in \mathcal{FL}$ from above proposition, so $(\mathcal{K}\mathcal{E}_{uc}^*)(sa) \leq (\mathcal{K}\mathcal{E}_{uc}^*)(s)$.

For later need, we define inductively another fuzzy subset $\check{\mathcal{K}}$ of E^* as follows:

$$\check{\mathcal{K}}(\epsilon) = 1;$$

$$\check{\mathcal{K}}(sa) = \begin{cases} \mathcal{K}(sa), & \text{if } a \in E_c \\ \check{\mathcal{K}}(s) \wedge \mathcal{L}(sa), & \text{if } a \in E_{uc} \end{cases}$$

for all $s \in E^*$ and $a \in E$.

We first present a compact formula for $\check{\mathcal{K}}$: $\check{\mathcal{K}} = \mathcal{K}\mathcal{E}_{uc}^* \cap \mathcal{L}$; furthermore, $\check{\mathcal{K}} \in \mathcal{FL}$.

We prove the first part by induction on the length of strings s . In the case of $s = \epsilon$, we see that $\check{\mathcal{K}}(\epsilon) = (\mathcal{K}\mathcal{E}_{uc}^* \cap \mathcal{L})(\epsilon) = 1$, so the basis step holds. Assume that $\check{\mathcal{K}}(s) = (\mathcal{K}\mathcal{E}_{uc}^* \cap \mathcal{L})(s)$ for all strings s with $|s| \leq n$. Now consider the case of sa , where $a \in E$. In the subcase of $a \in E_c$, we know that $\check{\mathcal{K}}(sa) = \mathcal{K}(sa)$ by definition. On the other hand, $(\mathcal{K}\mathcal{E}_{uc}^* \cap \mathcal{L})(sa) = \mathcal{K}(sa) \wedge \mathcal{L}(sa) = \mathcal{K}(sa)$. Consequently, $\check{\mathcal{K}}(sa) = (\mathcal{K}\mathcal{E}_{uc}^* \cap \mathcal{L})(sa)$ when $a \in E_c$. In the other subcase $a \in E_{uc}$, we have that

$$\begin{aligned} \check{\mathcal{K}}(sa) &= \check{\mathcal{K}}(s) \wedge \mathcal{L}(sa) \quad (\text{by definition}) \\ &= (\mathcal{K}\mathcal{E}_{uc}^* \cap \mathcal{L})(s) \wedge \mathcal{L}(sa) \quad (\text{by induction hypothesis}) \\ &= (\mathcal{K}\mathcal{E}_{uc}^*)(s) \wedge \mathcal{L}(s) \wedge \mathcal{L}(sa) \\ &= (\mathcal{K}\mathcal{E}_{uc}^*)(s) \wedge \mathcal{L}(sa), \quad \text{and} \\ (\mathcal{K}\mathcal{E}_{uc}^* \cap \mathcal{L})(sa) &= (\mathcal{K}\mathcal{E}_{uc}^*)(sa) \wedge \mathcal{L}(sa) \\ &= (\mathcal{K}\mathcal{E}_{uc}^*)(s) \wedge \mathcal{L}(sa). \end{aligned}$$

Hence we also have that $\check{\mathcal{K}}(sa) = (\mathcal{K}\mathcal{E}_{uc}^* \cap \mathcal{L})(sa)$ when $a \in E_{uc}$. The second part follows from the first part, since $\mathcal{K}$, $\mathcal{E}_{uc}^*$, and $\mathcal{L}$ are fuzzy languages.

Our intention of defining $\check{\mathcal{K}}$ is to represent $\mathcal{K}^\downarrow$; we establish the following theorem [CY05]: $\mathcal{K}^\downarrow = \check{\mathcal{K}}$. Thus, we have obtained the following characterizations of the least controllable fuzzy superlanguage $\mathcal{K}^\downarrow$ of $\mathcal{K}$.

We have:

$$\mathcal{K}^\downarrow = \bigcap_{\mathcal{M} \in C_{sup}(\mathcal{K})} \mathcal{M} = \mathcal{K}\mathcal{E}_{uc}^* \cap \mathcal{L}.$$

More explicitly, $\mathcal{K}^\downarrow$ can be defined inductively as follows: $\mathcal{K}^\downarrow(\epsilon) = 1;$

$$\mathcal{K}^\downarrow(sa) = \begin{cases} \mathcal{K}(sa), & \text{if } a \in E_c \\ \mathcal{K}^\downarrow(s) \wedge \mathcal{L}(sa), & \text{if } a \in E_{uc} \end{cases}$$

for all $s \in E^*$ and $a \in E$.

Using the above theorem, we can get the support of $\mathcal{K}^\downarrow$ from those of $\mathcal{K}$ and $\mathcal{L}$:

$$\text{supp}(\mathcal{K}^\downarrow) = \text{supp}(\mathcal{K})E_{uc}^* \cap \text{supp}(\mathcal{L}).$$

Note that $\text{supp}(\mathcal{E}_{uc}^*) = E_{uc}^*$ by definition.

Based on the above theorem, we now present an algorithm for the computation of $\mathcal{K}^\downarrow$ from automata H and G that generate $\mathcal{K}$ and $\mathcal{L}$, respectively.

Let $H = (Q_H, E, \delta_H, q_{0H})$ and $G = (Q, E, \delta, q_0)$.

Algorithm for $\mathcal{K}^\downarrow$:

Step 1). Build a fuzzy automaton that generates $\mathcal{K}\mathcal{E}_{uc}^*$ as follows. Add a new state q_a to Q_H . For all $q \in Q_H \cup \{q_a\}$ and all $a \in E_{uc}$, add an arc with label $a|1$ from q to q_a . Call the resulting automaton H_1 . More formally, $H_1 = (Q_1, E, \delta_1, q_{01})$ is given by

$Q_1 = Q_H \cup \{q_a\}$, $q_{01} = q_0$, and

$$\delta_1(p, a, q) = \begin{cases} \delta_H(p, a, q), & \text{if } p, q \in Q_H \\ 1, & \text{if } a \in E_{uc} \text{ and } q = q_a \\ 0, & \text{otherwise.} \end{cases}$$

Step 2). Take the product operation $H_1 \times G$. Then it follows immediately that $\mathcal{L}(H_1 \times G) = \mathcal{K}^\downarrow$.

The correctness of this algorithm is easily verified once one observes that Step 1) serves to the concatenation and the Kleene closure in the above theorem.

Now, we have the following fact. Let $\mathcal{K}_1, \mathcal{K}_2 \in \mathcal{FL}$, then:

- 1) If $\mathcal{K}_1 \subseteq \mathcal{K}_2$, then $\mathcal{K}_1^\downarrow \subseteq \mathcal{K}_2^\downarrow$.
- 2) $(\mathcal{K}_1 \cap \mathcal{K}_2)^\downarrow = \mathcal{K}_1^\downarrow \cap \mathcal{K}_2^\downarrow$.
- 3) $(\mathcal{K}_1 \cup \mathcal{K}_2)^\downarrow = \mathcal{K}_1^\downarrow \cup \mathcal{K}_2^\downarrow$.

The assertion 1) is clear by the definition of $\mathcal{K}^\downarrow$. The assertions 2) and 3) can be proved in the same way using induction on the length of strings, so we only prove 2). We have that $(\mathcal{K}_1 \cap \mathcal{K}_2)^\downarrow(\epsilon) = 1 = (\mathcal{K}_1^\downarrow \cap \mathcal{K}_2^\downarrow)(\epsilon)$. Suppose that 2) holds for all strings s with $|s| \leq n$, namely, $(\mathcal{K}_1 \cap \mathcal{K}_2)^\downarrow(s) = (\mathcal{K}_1^\downarrow \cap \mathcal{K}_2^\downarrow)(s)$. We now consider the case of sa . If $a \in E_c$, then we get that $(\mathcal{K}_1 \cap \mathcal{K}_2)^\downarrow(sa) = (\mathcal{K}_1 \cap \mathcal{K}_2)(sa) = \mathcal{K}_1(sa) \wedge \mathcal{K}_2(sa) = \mathcal{K}_1^\downarrow(sa) \wedge \mathcal{K}_2^\downarrow(sa) = (\mathcal{K}_1^\downarrow \cap \mathcal{K}_2^\downarrow)(sa)$. If $a \in E_{uc}$, then by induction hypothesis we have that

$$\begin{aligned} (\mathcal{K}_1 \cap \mathcal{K}_2)^\downarrow(sa) &= (\mathcal{K}_1 \cap \mathcal{K}_2)^\downarrow(s) \wedge \mathcal{L}(sa) \\ &= (\mathcal{K}_1^\downarrow \cap \mathcal{K}_2^\downarrow)(s) \wedge \mathcal{L}(sa) \\ &= \mathcal{K}_1^\downarrow(s) \wedge \mathcal{K}_2^\downarrow(s) \wedge \mathcal{L}(sa) \\ &= (\mathcal{K}_1^\downarrow(s) \wedge \mathcal{L}(sa)) \wedge (\mathcal{K}_2^\downarrow(s) \wedge \mathcal{L}(sa)) \\ &= \mathcal{K}_1^\downarrow(sa) \wedge \mathcal{K}_2^\downarrow(sa) \\ &= (\mathcal{K}_1^\downarrow \cap \mathcal{K}_2^\downarrow)(sa). \end{aligned}$$

Finally, recall that SCP mentioned above requires us to find a fuzzy supervisor S for the fuzzy DES G such that the behavior of controlled system lies between the minimal acceptable behavior $\mathcal{L}_a$ and the maximal legal behavior $\mathcal{L}_l$. We end this section with the solution to SCP. The following statements are equivalent [CY05]: (i) SCP is solvable; (ii) $\mathcal{L}_a \subseteq \mathcal{L}_l^\uparrow$; and (iii) $\mathcal{L}_a^\downarrow \subseteq \mathcal{L}_l$.

4.6 Neuro-Fuzzy Associative n -Functors

4.6.1 High-Resolution Functorial Agents

Recall from second Chapter that *multi-agent systems* have been successfully applied in numerous fields. For example, the emerging field of *cellular robotics* relates to building robots on modular basis: a robot (say a manipulator arm), is considered as a multi-agent system, and each of its constituent parts (arm segments) is regarded as an agent (with the joints describing constraints on the set of acceptable movements). The coordination of movement is thus a result of coordination involving an assembly of agents. It has been claimed that techniques similar to those used for modeling the multi-agent based eco-systems make it possible to accomplish complex movements with a minimum of computation. The head agent attempts to achieve the goal which has been set for it: if it can do it itself, it makes the movement and the system comes to a halt; if not, it brings in the neighboring agent, giving it goals which will help the head agent to get nearer to its objective. The process is repeated recursively, each agent trying to achieve the goals set for it by transmitting its desiderata to the neighboring agent.

This concept looks fine at the *low resolution level*. However, if we want a more sophisticated, high-resolution behavior of the arm manipulator, then we need to calculate a few derivatives of the nice and simple command ‘Move’. This would give us proper robot kinematics: velocities and accelerations. In addition, if we want some more realism, we need to include inertial, gravity and friction forces, as well as to compensate them by driving forces. In this way we come to the proper dynamical system. Even further, to make it a proper control system, we need to add some feedbacks.

In this subsection we present a *high-resolution agent model*, suitable for brain and cognition modeling. Our high-resolution agent model is a flexible and adaptive n -functorial organization. Its basic unit is a generic nonlinear MIMO-FAM functor $\mathcal{F}$, given by

$$\begin{array}{ccc} INPUT & \xrightarrow[\text{FAM}]{\mathcal{F}} & OUTPUT \\ \text{CAT} & & \text{CAT} \end{array} \quad (4.8)$$

where $INPUT_{CAT}$ and $OUTPUT_{CAT}$ are n -categories consisting of *adaptive fuzzy processes*, K.e., temporal fuzzy variables partitioned by overlapping membership functions, which are either standard triangular/trapesoidal functions, or Gaussian functions of the form $\mu(z) = \exp\left[\frac{-(z-m)^2}{2\sigma^2}\right]$, where

$z = z(t)$ denotes any fuzzy variable, while m and σ are its adjustable parameters, mean and standard deviation, respectively. The corresponding adaptive $n \times m$ -dimensional functor $\mathbb{R}^n \xrightarrow[\text{FAM}]{\mathcal{F}} \mathbb{R}^m$ denotes the FAM-functor, that can be exemplified by a (possibly sparse) matrix R^{ij} of fuzzy IF-THEN rules of the form:

IF x_1 is $\mu(x)_1^{\mathcal{K}}$ AND ... x_n is $\mu(x)_n^{\mathcal{K}}$ THEN y_1 is $\mu(y)_1^{\mathcal{J}}$ AND ... y_m is $\mu(y)_m^{\mathcal{J}}$ each with its associated *rule-importance weight* $w_{ij} \in [0, 1]$.

Given several FAM-functors of the form (4.8), we can perform their *associative composition* and *distributive fusion*:

$$\begin{array}{ccccc}
 X & \xrightarrow[\text{FAM}]{\mathcal{F}} & Y & \xrightarrow[\text{FAM}]{\mathcal{G}} & Z \\
 \text{CAT} & & \text{CAT} & & \text{CAT} \\
 & \uparrow \text{FAM} & & & \\
 & \mathcal{H} & & & \\
 & W & & & \\
 & \text{CAT} & & &
 \end{array}$$

Using composition and fusion of FAM-functors, we can form hierarchical networks of nonlinear MIMO systems, for modeling human behavior real-life games.

Both the rule-weights and Gaussian parameters (means and standard deviations) are adjusted using two forms of unsupervised trial-and-error learning: *Hebbian learning*, minimizing the error and *reinforcement learning*, maximizing the reward (see second Chapter). We use both types of learning only as a minor parameter-adjusting tool, secondary to the primary fuzzy-logic *instruction*, setting-up the system's structure (K.e., FAM-functor configuration). The learning is performed in discrete time steps, according to the general self-organized rule:

$$\text{new value} = \text{old value} + \text{innovation}$$

The local learning dynamics is performed as a combination of the two different ways:

1. Immediate, feedback-based, *Hebbian learning* rule [Heb49], occurring after each input-output system pass, comparing achieved output execution with the desired one and not considering the long-term rewards. Here the system learns its *optimal performance*:

$$\text{innovation} = |\text{desired output} - \text{achieved output}|$$

e.G., in case of the tennis game, this means the proper execution (K.e., precision, speed and spin) of the shot, without regard to the winning-points. It follows the simple tactic: 'Let me do my best shot, regardless of the opponent's actions'.

2. Long-term, reward-based, *reinforcement learning* rule [SB98], occurring only after winning points and remembering the rewards. Here the system learns its *optimal policy*:

$$innovation = |reward - penalty|$$

In the tennis case, this means only the winning-point, no matter what was the actual execution of the shot. It follows the competitive tactics: ‘The goal is to beat the opponent, no matter how (within the rules)’.

4.6.2 Application to Real-Life Games

As a first application of our high-resolution functorial agent, we will formulate a continuous-time *attack* (AT) and *counter-attack* (CA) model for the *tennis game*, with discrete-time learning. Recall that tennis represents a competitive real-life situation within fully controlled conditions, depending on the individual’s technical², tactical³ and strategic⁴ skills as well as operational psycho-physical strengths. Here, we will formulate an adaptive, fuzzy-dynamics AT & CA, tennis performance model. Using compositions and fusions of FAM-functors (as described above), we will design a generic simulator for counter-attack and attack performance dynamics, based on the tennis game. We have chosen the tennis game as it allows: (K) complete analysis in every detail, (ii) full experimental validation, and (iii) training both in real-life situations and in a machine-learning environment.

Attack Model: Tennis Serve

A. Simple Attack: Serve Only. The simple AT-dynamics is represented by a single FAM functor

$$\underset{CAT}{TARGET} \xrightarrow[\underset{FAM}{\mathcal{F}^{AT}}]{} \underset{CAT}{ATTACK}$$

In the case of simple tennis serve, this AT-scenario reads

$$\underset{OPPONENT-IN}{O \ni o_m} \xrightarrow{\mathcal{F}^{AT}} \underset{SERVE-OUT}{SR \ni sr_n}$$

² *Technical skills* in sports are judged by the *biomechanical correctness of movements*.

³ *Tactical skills* in sports include ‘winners’, ‘changes of tempo’, and ‘applying pressure’.

⁴ *Strategic skills* in sports include ‘planning ahead’ and ‘weighting the relative importance of points’ along the match.

where the two n -categories, $O_{\dim=2} \ni o_m$ and $SR_{\dim=3} \ni sr_n$, contain the fuzzy variables $\{o_m = o_m(t)\}$ and $\{sr_n = sr_n(t)\}$, respectively opponent-related (target information) and serve-related, partitioned by overlapping Gaussians, $\mu(z) = \exp \left[\frac{-(z-m)^2}{2\sigma^2} \right]$, and defined as:

$$\begin{array}{l} O_{\text{OPPONENT-IN}} : \begin{array}{l} o_1 = \text{Opp.Posit.Left.Right} : (\text{center}, \text{medium}, \text{wide}), \\ o_2 = \text{Opp.Antcp.Left.Right} : (\text{runCenter}, \text{stay}, \text{runWide}), \end{array} \\ \\ SR_{\text{SERVE-OUT}} : \begin{array}{l} sr_1 = 1.\text{Serve.Speed} : (\text{low}, \text{medium}, \text{high}) \\ sr_2 = 2.\text{Serve.Spin} : (\text{low}, \text{medium}, \text{high}) \\ sr_3 = 3.\text{Serve.Placement} : (\text{center}, \text{medium}, \text{wide}) \end{array} \end{array}$$

In the fuzzy-matrix form this simple serve reads

$$\begin{array}{c} O: \text{OPPONENT-IN} \\ \left[\begin{array}{l} o_1 = \text{Opp.Posit.Left.Right} \\ o_2 = \text{Opp.Anticip.Left.Right} \end{array} \right] \end{array} \xrightarrow{\mathcal{F}^{AT}} \begin{array}{c} SR: \text{SERVE-OUT} \\ \left[\begin{array}{l} sr_1 = 1.\text{Serve.Speed} \\ sr_2 = 2.\text{Serve.Spin} \\ sr_3 = 3.\text{Serve.Place} \end{array} \right] \end{array}$$

B. Attack-Maneuver: Serve-Volley. The generic advanced AT-dynamics is given by a composition of FAM functors

$$\begin{array}{c} \text{TARGET} \\ \text{CAT} \end{array} \xrightarrow[\text{FAM}]{\mathcal{F}^{AT}} \begin{array}{c} \text{ATTACK} \\ \text{CAT} \end{array} \xrightarrow[\text{FAM}]{\mathcal{G}^{AT}} \begin{array}{c} \text{MANEUVER} \\ \text{CAT} \end{array}$$

In the case of advanced tennis serve, this AT-scenario reads

$$\begin{array}{c} O \ni o_m \\ \text{OPPONENT-IN} \end{array} \xrightarrow[\text{SERVE-OUT}]{\mathcal{F}^{AT}} \begin{array}{c} SR \ni sr_n \\ \text{SERVE-OUT} \end{array} \xrightarrow[\text{RUN-VOLEY}]{\mathcal{G}^{AT}} \begin{array}{c} RV \ni rv_p \\ \text{RUN-VOLEY} \end{array}$$

where the new n -category, $RV_{\dim=2} \ni rv_p$, contains the opponent-anticipation driven volley-maneuver, expressed by fuzzy variables $\{rv_p = rv_p(t)\}$, partitioned by overlapping Gaussians and given by:

$$\begin{array}{l} RV_{\text{RUN-VOLEY}} : \begin{array}{l} rv_1 = \text{RV.For} : (\text{baseLine}, \text{center}, \text{netClose}) \\ rv_2 = \text{RV.L.R.} : (\text{left}, \text{center}, \text{right}) \end{array} \end{array}$$

In the fuzzy-matrix form this advanced serve reads

$$\begin{array}{c} O: \text{OPPONENT-IN} \\ \left[\begin{array}{l} o_1 = \text{Opp.Posit.L.R.} \\ o_2 = \text{Opp.Anticip.L.R.} \end{array} \right] \end{array} \xrightarrow{\mathcal{F}^{AT}} \begin{array}{c} SR: \text{SERVE-OUT} \\ \left[\begin{array}{l} sr_1 = 1.\text{Serve.Speed} \\ sr_2 = 2.\text{Serve.Spin} \\ sr_3 = 3.\text{Serve.Place} \end{array} \right] \end{array} \xrightarrow{\mathcal{G}^{AT}} \begin{array}{c} RV: \text{RUN-VOLEY} \\ \left[\begin{array}{l} rv_1 = \text{RV.For} \\ rv_2 = \text{RV.L.R.} \end{array} \right] \end{array}$$

Counter–Attack Model: Tennis Return

A. Simple Return. The simple CA–dynamics reads:

$$\begin{array}{ccccc} \textit{ATTACK} & \xrightarrow[\textit{FAM}]{\mathcal{F}^{CA}} & \textit{MANEUVER} & \xrightarrow[\textit{FAM}]{\mathcal{G}^{CA}} & \textit{RESPONSE} \\ \textit{CAT} & & \textit{CAT} & & \textit{CAT} \end{array}$$

In the case of simple tennis return, this CA–scenario consists purely of conditioned–reflex reaction, no decision process is involved, so it reads:

$$\begin{array}{ccccc} B \ni b_K & \xrightarrow{\mathcal{F}^{CA}} & R \ni r_{\mathcal{J}} & \xrightarrow{\mathcal{G}^{CA}} & S \ni s_k \\ \textit{BALL-IN} & & \textit{RUNNING} & & \textit{SHOT-OUT} \end{array}$$

where the n –categories $B_{\dim=5} \ni b_K$, $R_{\dim=3} \ni r_{\mathcal{J}}$, $S_{\dim=4} \ni s_k$, contain the fuzzy variables $\{b_K = b_K(t)\}$, $\{r_{\mathcal{J}} = r_{\mathcal{J}}(t)\}$ and $\{s_k = s_k(t)\}$, respectively defining the ball inputs, our player’s running maneuver and his shot–response, K.e.,

$$\begin{array}{ccc} \begin{array}{l} \textit{B: BALL-IN} \\ b_1 = \textit{Dist.L.R.} \\ b_2 = \textit{Dist.F.B.} \\ b_3 = \textit{Dist.Vert} \\ b_4 = \textit{Speed} \\ b_5 = \textit{Spin} \end{array} & \xrightarrow{\mathcal{F}^{CA}} & \begin{array}{l} \textit{R: RUNNING} \\ r_1 = \textit{Run.L.R.} \\ r_2 = \textit{Run.F.B.} \\ r_3 = \textit{Run.Vert} \end{array} \xrightarrow{\mathcal{G}^{CA}} \begin{array}{l} \textit{S: SHOT-OUT} \\ s_1 = \textit{Backhand} \\ s_2 = \textit{Forehand} \\ s_3 = \textit{Voley} \\ s_4 = \textit{Smash} \end{array} \end{array}$$

Here, the existence of efficient weapons within the $S_{\textit{SHOT-OUT}}$ arsenal–space, namely $s_k(t) : s_1 = \textit{Backhand}, s_2 = \textit{Forehand}, s_3 = \textit{Voley}$ and $s_4 = \textit{Smash}$, is assumed.

The universes of discourse for the fuzzy variables $\{b_K(t)\}$, $\{r_{\mathcal{J}}(t)\}$ and $\{s_k(t)\}$, partitioned by overlapping Gaussians, are defined respectively as:

$$\begin{array}{l} \textit{B}_{\textit{BALL-IN}} : \\ b_1 = \textit{Dist.L.R.} : (\textit{veryLeft}, \textit{left}, \textit{center}, \textit{right}, \textit{veryRight}), \\ b_2 = \textit{Dist.F.B.} : (\textit{baseLine}, \textit{center}, \textit{netClose}), \\ b_3 = \textit{Dist.Vert} : (\textit{low}, \textit{medium}, \textit{high}), \\ b_4 = \textit{Speed} : (\textit{low}, \textit{medium}, \textit{high}), \\ b_5 = \textit{Spin} : (\textit{highTopSpin}, \textit{lowTopSpin}, \textit{flat}, \\ \quad \textit{lowBackSpin}, \textit{highBackSpin}). \end{array}$$

$$\begin{array}{l} \textit{R}_{\textit{RUNNING}} : \\ r_1 = \textit{Run.L.R.} : (\textit{veryLeft}, \textit{left}, \textit{center}, \textit{right}, \textit{veryRight}), \\ r_2 = \textit{Run.F.B.} : (\textit{closeFront}, \textit{front}, \textit{center}, \textit{back}, \textit{farBack}), \\ r_3 = \textit{Run.Vert} : (\textit{squat}, \textit{normal}, \textit{jump}). \end{array}$$

$$\begin{array}{l} \textit{S}_{\textit{SHOT-OUT}} : \\ s_1 = \textit{Backhand} : (\textit{low}, \textit{medium}, \textit{high}), \\ s_2 = \textit{Forehand} : (\textit{low}, \textit{medium}, \textit{high}), \\ s_3 = \textit{Voley} : (\textit{backhand}, \textit{block}, \textit{forehand}), \\ s_4 = \textit{Smash} : (\textit{low}, \textit{medium}, \textit{high}). \end{array}$$

B. Advanced Return. The advanced CA-dynamics includes both the information about the opponent and (either conscious or subconscious) decision making. This generic CA-scenario is formulated as the following composition + fusion of FAM functors:

$$\begin{array}{ccccccc}
 \underset{CAT}{ATTACK} & \xrightarrow[\underset{FAM}{\mathcal{F}^{CA}}]{} & \underset{CAT}{MANEUVER} & \xrightarrow[\underset{FAM}{\mathcal{G}^{CA}}]{} & \underset{CAT}{DECISION} & \xrightarrow[\underset{FAM}{\mathcal{H}^{CA}}]{} & \underset{CAT}{RESPONSE} \\
 & & & & \uparrow \mathcal{K}^{CA} & & \\
 & & & & \underset{CAT}{TARGET} & &
 \end{array}$$

where we have added two new n -categories, $\underset{CAT}{TARGET}$ and $\underset{CAT}{DECISION}$, respectively containing information about the opponent as a target, as well as our own aiming decision processes. In the case of advanced tennis return, this reads:

$$\begin{array}{ccccccc}
 \underset{BALL-IN}{B \ni b_K} & \xrightarrow{\mathcal{F}^{CA}} & \underset{RUNNING}{R \ni r_J} & \xrightarrow{\mathcal{G}^{CA}} & \underset{DECISION}{D \ni d_l} & \xrightarrow{\mathcal{H}^{CA}} & \underset{SHOT-OUT}{S \ni s_k} \\
 & & & & \uparrow \mathcal{K}^{CA} & & \\
 & & & & \underset{OPPONENT-IN}{O \ni o_m} & &
 \end{array}$$

where the two additional n -categories, $O_{\dim=4} \ni o_m$ and $D_{\dim=5} \ni d_l$, contain the fuzzy variables $\{o_m = o_m(t)\}$ and $\{d_l = d_l(t)\}$, respectively defining the opponent-related target information and the aim-related decision processes, both partitioned by overlapping Gaussians and defined as:

$$\underset{OPPONENT-IN}{O} : \begin{cases} o_1 = Opp.Posit.L.R. : (left, center, right), \\ o_2 = Opp.Posit.F.B. : (netClose, center, baseLine), \\ o_3 = Opp.Anticip.L.R. : (runLeft, stay, runRight), \\ o_4 = Opp.Anticip.F.B. : (runNet, stay, runBase). \end{cases}$$

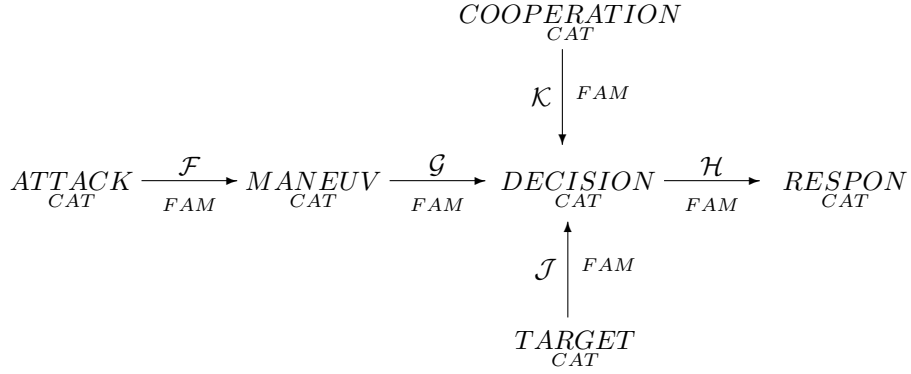
$$\underset{DECISION}{D} : \begin{cases} d_1 = Aim.L.R. : (left, center, right), \\ d_2 = Aim.F.B. : (netClose, center, baseLine), \\ d_3 = Aim.Vert : (low, medium, high), \\ d_4 = Aim.Speed : (low, medium, high), \\ d_5 = Aim.Spin : (highTopSpin, lowTopSpin, noSpin, \\ \quad lowBackSpin, highBackSpin). \end{cases}$$

The corresponding fuzzy-matrices read:

$$\begin{array}{ccc}
\begin{array}{l} B: BALL-IN \\ \left[\begin{array}{l} b_1 = Dist.L.R. \\ b_2 = Dist.F.B. \\ b_3 = Dist.Vert \\ b_4 = Speed \\ b_5 = Spin \end{array} \right] , & \begin{array}{l} R: RUNNING \\ \left[\begin{array}{l} r_1 = Run.L.R. \\ r_2 = Run.F.B. \\ r_3 = Run.Vert \end{array} \right] , & \begin{array}{l} D: DECISION \\ \left[\begin{array}{l} d_1 = Aim.L.R. \\ d_2 = Aim.F.B. \\ d_3 = Aim.Vert \\ d_4 = Aim.Speed \\ d_5 = Aim.Spin \end{array} \right] , \\ \\ \begin{array}{l} O: OPPONENT-IN \\ \left[\begin{array}{l} o_1 = Opp.Posit.L.R. \\ o_2 = Opp.Posit.F.B. \\ o_3 = Opp.Anticip.L.R. \\ o_4 = Opp.Anticip.F.B. \end{array} \right] , & \begin{array}{l} S: SHOT-OUT \\ \left[\begin{array}{l} s_1 = Backhand \\ s_2 = Forehand \\ s_3 = Voley \\ s_4 = Smash \end{array} \right] .
\end{array}
\end{array}$$

4.6.3 Generic Functorial FAM–Agents

Finally, we briefly describe a generic functorial *FAM–agent* as a following complex adaptive system, designed from five adaptive FAM–functors and six continuous–time + discrete–time n –categories:



Each generic FAM–agent is supposed to communicate with other FAM–agents and/or environment only by its inputs and outputs. It is both a feedback control system and an expert system, implementing: (K) feedbacks, (ii) instructions and (iii) common sense. It is a high–resolution human performance predictor–corrector system, which has to be designed in each specific case by consulting an expert from the field.

A *smart multi–agent organization* represents a brain–like sensory–motor system–of–systems, including hierarchical networks of generic functorial FAM–agents, with the unique input and output n –categories.

Appendix

5.1 Universal Associative Language of Categories and Functors

In modern mathematical sciences whenever one defines a new class of mathematical objects, one proceeds almost in the next breath to say what kinds of maps between objects will be considered [Swi75, II06b]. A general framework for dealing with situations where we have some *objects* and *maps between objects*, like sets and functions, vector spaces and linear operators, points in a space and paths between points, etc. – gives the modern metalanguage of categories and functors. Categories are mathematical universes and functors are ‘projectors’ from one universe onto another. For this reason, in this book we extensively use this associative language, mainly following its founder, S. MacLane [MacL71].

5.1.1 Maps

Notes from Set Theory

Given a map (or, a function) $f : A \rightarrow B$, the set A is called the *domain* of f , and denoted $\text{Dom } f$. The set B is called the *codomain* of f , and denoted $\text{Cod } f$. The codomain is not to be confused with the *range* of $f(A)$, which is in general only a subset of B .

A map $f : X \rightarrow Y$ is called *injective* or 1–1 or an *injection* if for every y in the codomain Y there is at most one x in the domain X with $f(x) = y$. Put another way, given x and x' in X , if $f(x) = f(x')$, then it follows that $x = x'$. A map $f : X \rightarrow Y$ is called *surjective* or *onto* or a *surjection* if for every y in the codomain $\text{Cod } f$ there is at least one x in the domain X with $f(x) = y$. Put another way, the *range* $f(X)$ is equal to the codomain Y . A map is *bijective* iff it is both injective and surjective. Injective functions are called the *monomorphisms*, and surjective functions are called the *epimorphisms* in the *category of sets* (see below).

Two main classes of maps (or, functions) that we will use in this book are: (i) continuous maps (denoted as C^0 -class), and (ii) smooth or differentiable maps (denoted as C^∞ -class). The former class is the core of topology, the latter of differential geometry. They are both used in the core concept of manifold.

A *relation* is any subset of a *Cartesian product* (see below). By definition, an *equivalence relation* α on a set X is a relation which is *reflexive*, *symmetrical* and *transitive*, i.e., relation that satisfies the following three conditions:

1. *Reflexivity*: each element $x \in X$ is equivalent to itself, i.e., $x\alpha x$,
2. *Symmetry*: for any two elements $x, x' \in X$, $x\alpha x'$ implies $x'\alpha x$, and
3. *Transitivity*: $a \leq b$ and $b \leq c$ implies $a \leq c$.

Similarly, a relation $\leq$ defines a *partial order* on a set S if it has the following properties:

1. *Reflexivity*: $a \leq a$ for all $a \in S$,
2. *Antisymmetry*: $a \leq b$ and $b \leq a$ implies $a = b$, and
3. *Transitivity*: $a \leq b$ and $b \leq c$ implies $a \leq c$.

A *partially ordered set* (or *poset*) is a set taken together with a partial order on it. Formally, a partially ordered set is defined as an ordered pair $P = (X, \leq)$, where X is called the *ground set* of P and $\leq$ is the partial order of P .

Notes From Calculus

Maps

Recall that a *map* (or, *function*) f is a *rule* that assigns to each element x in a set A exactly one element, called $f(x)$, in a set B . A map could be thought of as a *machine* $[[f]]$ with x -input (the *domain* of f is the set of all possible inputs) and $f(x)$ -output (the *range* of f is the set of all possible outputs) [Stu99]

$$x \rightarrow [[f]] \rightarrow f(x)$$

There are four possible ways to represent a function (or map): (i) verbally (by a description in words); (ii) numerically (by a table of values); (iii) visually (by a graph); and (iv) algebraically (by an explicit formula). The most common method for visualizing a function is its *graph*. If f is a function with domain A , then its graph is the set of ordered input-output pairs

$$\{(x, f(x)) : x \in A\}.$$

A generalization of the graph concept is a concept of a *cross-section of a fibre bundle*, which is one of the core geometrical objects for dynamics of complex systems.

Algebra of Maps

Let f and g be maps with domains A and B . Then the maps $f + g$, $f - g$, fg , and f/g are defined as follows [Stu99]

$$\begin{aligned} (f + g)(x) &= f(x) + g(x) & \text{domain} &= A \cap B, \\ (f - g)(x) &= f(x) - g(x) & \text{domain} &= A \cap B, \\ (fg)(x) &= f(x)g(x) & \text{domain} &= A \cap B, \\ \left(\frac{f}{g}\right)(x) &= \frac{f(x)}{g(x)} & \text{domain} &= \{x \in A \cap B : g(x) \neq 0\}. \end{aligned}$$

Compositions of Maps

Given two maps f and g , the composite map $f \circ g$ (also called the *composition* of f and g) is defined by

$$(f \circ g)(x) = f(g(x)).$$

The $(f \circ g)$ -machine is composed of the g -machine (first) and then the f -machine [Stu99],

$$x \rightarrow [[g]] \rightarrow g(x) \rightarrow [[f]] \rightarrow f(g(x))$$

For example, suppose that $y = f(u) = \sqrt{u}$ and $u = g(x) = x^2 + 1$. Since y is a function of u and u is a function of x , it follows that y is ultimately a function of x . We calculate this by substitution

$$y = f(u) = f \circ g = f(g(x)) = f(x^2 + 1) = \sqrt{x^2 + 1}.$$

The Chain Rule

If f and g are both differentiable (or smooth, i.e., C^∞) maps and $h = f \circ g$ is the composite map defined by $h(x) = f(g(x))$, then h is differentiable and h' is given by the product [Stu99]

$$h'(x) = f'(g(x))g'(x).$$

In Leibniz notation, if $y = f(u)$ and $u = g(x)$ are both differentiable maps, then

$$\frac{dy}{dx} = \frac{dy}{du} \frac{du}{dx}.$$

The reason for the name *chain rule* becomes clear if we add another link to the chain. Suppose that we have one more differentiable map $x = h(t)$. Then, to calculate the derivative of y with respect to t , we use the chain rule twice,

$$\frac{dy}{dt} = \frac{dy}{du} \frac{du}{dx} \frac{dx}{dt}.$$

Integration and Change of Variables

1–1 continuous (i.e., C^0) map T with a nonzero *Jacobian* $\left| \frac{\partial(x, \dots)}{\partial(u, \dots)} \right|$ that maps a region S onto a region R , (see [Stu99]) we have the following substitution formulas:

1. for a single integral,

$$\int_R f(x) dx = \int_S f(x(u)) \frac{\partial x}{\partial u} du,$$

2. for a double integral,

$$\iint_R f(x, y) dA = \iint_S f(x(u, v), y(u, v)) \left| \frac{\partial(x, y)}{\partial(u, v)} \right| dudv,$$

3. for a triple integral,

$$\iiint_R f(x, y, z) dV = \iiint_S f(x(u, v, w), y(u, v, w), z(u, v, w)) \left| \frac{\partial(x, y, z)}{\partial(u, v, w)} \right| dudvdw$$

4. similarly for n -tuple integrals.

Notes from General Topology

Topology is a kind of *abstraction* of Euclidean geometry, and also a natural framework for the study of *continuity*.¹ Euclidean geometry is abstracted by regarding triangles, circles, and squares as being the same basic object. Continuity enters because in saying this one has in mind a *continuous deformation* of a triangle into a square or a circle, or any arbitrary shape. On the other hand, a disk with a hole in the center is topologically different from a circle or a square because one cannot create or destroy holes by continuous deformations. Thus using topological methods one does not expect to be able to identify a geometrical figure as being a triangle or a square. However, one does expect to be able to detect the presence of gross features such as holes or the fact that the figure is made up of two disjoint pieces etc. In this way topology produces theorems that are usually qualitative in nature – they may assert, for example, the existence or non-existence of an object. They will not, in general, give the means for its construction [Nas83].

¹ Intuitively speaking, a function $f : \mathbb{R} \rightarrow \mathbb{R}$ is continuous near a point x in its domain if its value does not jump there. That is, if we just take δx to be small enough, the two function values $f(x)$ and $f(x + \delta x)$ should approach each other arbitrarily closely. In more rigorous terms, this leads to the following definition: A function $f : \mathbb{R} \rightarrow \mathbb{R}$ is continuous at $x \in \mathbb{R}$ if for all $\epsilon > 0$, there exists a $\delta > 0$ such that for all $y \in \mathbb{R}$ with $|y - x| < \delta$, we have that $|f(y) - f(x)| < \epsilon$. The whole function is called continuous if it is continuous at every point x .

Topological Space

Study of topology starts with the fundamental notion of *topological space*. Let X be any set and $Y = \{X_\alpha\}$ denote a collection, finite or infinite of subsets of X . Then X and Y form a topological space provided the X_α and Y satisfy:

1. Any finite or infinite subcollection $\{Z_\alpha\} \subset X_\alpha$ has the property that $\cup Z_\alpha \in Y$, and
2. Any finite subcollection $\{Z_{\alpha_1}, \dots, Z_{\alpha_n}\} \subset X_\alpha$ has the property that $\cap Z_{\alpha_i} \in Y$.

The set X is then called a topological space and the X_α are called *open sets*. The choice of Y satisfying (2) is said to give a topology to X .

Given two topological spaces X and Y , a *function* (or, a *map*) $f : X \rightarrow Y$ is *continuous* if the inverse image of an open set in Y is an open set in X .

The main general idea in topology is to study spaces which can be continuously deformed into one another, namely the idea of *homeomorphism*. If we have two topological spaces X and Y , then a map $f : X \rightarrow Y$ is called a homeomorphism iff

1. f is continuous (C^0), and
2. There exists an inverse of f , denoted f^{-1} , which is also continuous.

Definition (2) implies that if f is a homeomorphism then so is f^{-1} . Homeomorphism is the main topological example of *reflexive*, *symmetrical* and *transitive relation*, i.e., *equivalence relation*. Homeomorphism divides all topological spaces up into *equivalence classes*. In other words, a pair of topological spaces, X and Y , belong to the same equivalence class if they are homeomorphic.

The second example of topological equivalence relation is *homotopy*. While homeomorphism generates equivalence classes whose members are topological spaces, homotopy generates equivalence classes whose members are continuous (C^0) maps. Consider two continuous maps $f, g : X \rightarrow Y$ between topological spaces X and Y . Then the map f is said to be *homotopic* to the map g if f can be continuously deformed into g (see below for the precise definition of homotopy). Homotopy is an equivalence relation which divides the space of continuous maps between two topological spaces into equivalence classes [Nas83].

Another important notions in topology are *covering*, *compactness* and *connectedness*. Given a family of sets $\{X_\alpha\} = X$ say, then X is a *covering* of another set Y if $\cup X_\alpha$ contains Y . If all the X_α happen to be open sets the covering is called an *open covering*. Now consider the set Y and all its possible open coverings. The set Y is *compact* if for every open covering $\{X_\alpha\}$ with $\cup X_\alpha \supset Y$ there always exists a finite subcovering $\{X_1, \dots, X_n\}$ of Y with $X_1 \cup \dots \cup X_n \supset Y$. Again, we define a set Z to be *connected* if it cannot be written as $Z = Z_1 \cup Z_2$, where Z_1 and Z_2 are both open and $Z_1 \cap Z_2$ is an empty set.

Let $A_1, A_2, \dots, A_n$ be closed subspaces of a topological space X such that $X = \bigcup_{i=1}^n A_i$. Suppose $f_i : A_i \rightarrow Y$ is a function, $1 \leq i \leq n$, iff

$$f_i|_{A_i \cap A_j} = f_j|_{A_i \cap A_j}, 1 \leq i, j \leq n. \quad (5.1)$$

In this case f is continuous iff each f_i is. Using this procedure we can define a C^0 -function $f : X \rightarrow Y$ by cutting up the space X into closed subsets A_i and defining f on each A_i separately in such a way that $f|_{A_i}$ is obviously continuous; we then have only to check that the different definitions agree on the *overlaps* $A_i \cap A_j$.

The *universal property of the Cartesian product*: let $p_X : X \times Y \rightarrow X$, and $p_Y : X \times Y \rightarrow Y$ be the *projections* onto the first and second factors, respectively. Given any pair of functions $f : Z \rightarrow X$ and $g : Z \rightarrow Y$ there is a unique function $h : Z \rightarrow X \times Y$ such that $p_X \circ h = f$, and $p_Y \circ h = g$. Function h is continuous iff both f and g are. This property characterizes X/α up to homeomorphism. In particular, to check that a given function $h : Z \rightarrow X$ is continuous it will suffice to check that $p_X \circ h$ and $p_Y \circ h$ are continuous.

The *universal property of the quotient*: let α be an equivalence relation on a topological space X , let X/α denote the *space of equivalence classes* and $p_\alpha : X \rightarrow X/\alpha$ the *natural projection*. Given a function $f : X \rightarrow Y$, there is a function $f' : X/\alpha \rightarrow Y$ with $f' \circ p_\alpha = f$ iff $x\alpha x'$ implies $f(x) = f(x')$, for all $x \in X$. In this case f' is continuous iff f is. This property characterizes X/α up to homeomorphism.

Homotopy

Now we return to the fundamental notion of homotopy. Let I be a compact unit interval $I = [0, 1]$. A *homotopy* from X to Y is a continuous function $F : X \times I \rightarrow Y$. For each $t \in I$ one has $F_t : X \rightarrow Y$ defined by $F_t(x) = F(x, t)$ for all $x \in X$. The functions F_t are called the ‘stages’ of the homotopy. If $f, g : X \rightarrow Y$ are two continuous maps, we say f is homotopic to g , and write $f \simeq g$, if there is a homotopy $F : X \times I \rightarrow Y$ such that $F_0 = f$ and $F_1 = g$. In other words, f can be continuously deformed into g through the stages F_t . If $A \subset X$ is a subspace, then F is a homotopy relative to A if $F(a, t) = F(a, 0)$, for all $a \in A, t \in I$.

The homotopy relation $\simeq$ is an equivalence relation. To prove that we have $f \simeq f$ is obvious; take $F(x, t) = f(x)$, for all $x \in X, t \in I$. If $f \simeq g$ and F is a homotopy from f to g , then $G : X \times I \rightarrow Y$ defined by $G(x, t) = F(x, 1 - t)$, is a homotopy from g to f , i.e., $g \simeq f$. If $f \simeq g$ with homotopy F and $g \simeq h$ with homotopy G , then $f \simeq h$ with homotopy H defined by

$$H(x, t) = \begin{cases} F(x, t), & 0 \leq t \leq 1/2 \\ G(x, 2t - 1), & 1/2 \leq t \leq 1 \end{cases}.$$

To show that H is continuous we use the relation (5.1).

In this way, the set of all C^0 -functions $f : X \rightarrow Y$ between two topological spaces X and Y , called the *function space* and denoted by Y^X , is partitioned

into equivalence classes under the relation $\simeq$. The equivalence classes are called *homotopy classes*, the homotopy class of f is denoted by $[f]$, and the set of all homotopy classes is denoted by $[X; Y]$.

If α is an equivalence relation on a topological space X and $F : X \times I \rightarrow Y$ is a homotopy such that each stage F_t factors through X/α , i.e., $x\alpha x'$ implies $F_t(x) = F_t(x')$, then F induces a homotopy $F' : (X/\alpha) \times I \rightarrow Y$ such that $F' \circ (p_\alpha \times 1) = F$.

Homotopy theory has a range of applications of its own, outside topology and geometry, as for example in proving Cauchy theorem in complex variable theory, or in solving nonlinear equations of artificial neural networks.

A *pointed set* (S, s_0) is a set S together with a distinguished point $s_0 \in S$. Similarly, a *pointed topological space* (X, x_0) is a space X together with a distinguished point $x_0 \in X$. When we are concerned with pointed spaces $(X, x_0), (Y, y_0)$, etc., we always require that all functions $f : X \rightarrow Y$ shell preserve base points, i.e., $f(x_0) = y_0$, and that all homotopies $F : X \times I \rightarrow Y$ be relative to the base point, i.e., $F(x_0, t) = y_0$, for all $t \in I$. We denote the homotopy classes of base point-preserving functions by $[X, x_0; Y, y_0]$ (where homotopies are relative to x_0). $[X, x_0; Y, y_0]$ is a pointed set with base point f_0 , the constant function: $f_0(x) = y_0$, for all $x \in X$.

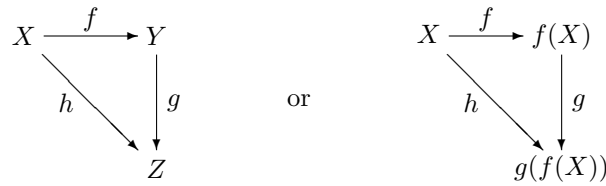
A *path* $\gamma(t)$ from x_0 to x_1 in a topological space X is a continuous map $\gamma : I \rightarrow X$ with $\gamma(0) = x_0$ and $\gamma(1) = x_1$. Thus X^I is the space of all paths in X with the compact-open topology. We introduce a relation $\sim$ on X by saying $x_0 \sim x_1$ iff there is a path $\gamma : I \rightarrow X$ from x_0 to x_1 . $\sim$ is clearly an equivalence relation, and the set of equivalence classes is denoted by $\pi_0(X)$. The elements of $\pi_0(X)$ are called the *path components*, or *0-components* of X . If $\pi_0(X)$ contains just one element, then X is called *path connected*, or *0-connected*. A *closed path*, or *loop* in X at the point x_0 is a path $\gamma(t)$ for which $\gamma(0) = \gamma(1) = x_0$. The *inverse loop* $\gamma^{-1}(t)$ based at $x_0 \in X$ is defined by $\gamma^{-1}(t) = \gamma(1 - t)$, for $0 \leq t \leq 1$. The *homotopy of loops* is the particular case of the above defined homotopy of continuous maps.

If (X, x_0) is a pointed space, then we may regard $\pi_0(X)$ as a pointed set with the 0-component of x_0 as a base point. We use the notation $\pi_0(X, x_0)$ to denote $p_0(X, x_0)$ thought of as a pointed set. If $f : X \rightarrow Y$ is a map then f sends 0-components of X into 0-components of Y and hence defines a function $\pi_0(f) : \pi_0(X) \rightarrow \pi_0(Y)$. Similarly, a base point-preserving map $f : (X, x_0) \rightarrow (Y, y_0)$ induces a map of pointed sets $\pi_0(f) : \pi_0(X, x_0) \rightarrow \pi_0(Y, y_0)$. In this way defined π_0 represents a ‘functor’ from the ‘category’ of topological (point) spaces to the underlying category of (point) sets (see the next subsection).

Combination of topology and calculus gives differential topology, or differential geometry.

Commutative Diagrams

The *category theory* (see below) was born with an observation that many properties of mathematical systems can be unified and simplified by a presentation with *commutative diagrams of arrows* [MacL71]. Each arrow $f : X \rightarrow Y$ represents a function (i.e., a map, transformation, operator); that is, a source (domain) set X , a target (codomain) set Y , and a rule $x \mapsto f(x)$ which assigns to each element $x \in X$ an element $f(x) \in Y$. A typical diagram of sets and functions is



This diagram is *commutative* iff $h = g \circ f$, where $g \circ f$ is the usual composite function $g \circ f : X \rightarrow Z$, defined by $x \mapsto g(f(x))$.

Similar commutative diagrams apply in other mathematical, physical and computing contexts; e.g., in the ‘category’ of all topological spaces, the letters X, Y , and Z represent topological spaces while f, g , and h stand for continuous maps. Again, in the category of all groups, X, Y , and Z stand for groups, f, g , and h for homomorphisms.

Less formally, composing maps is like following directed paths from one object to another (e.g., from set to set). In general, a diagram is commutative iff any two paths along arrows that start at the same point and finish at the same point yield the same ‘homomorphism’ via compositions along successive arrows. Commutativity of the whole diagram follows from commutativity of its triangular components (depicting a ‘commutative flow’, see Figure 5.1). Study of commutative diagrams is popularly called ‘diagram chasing’, and provides a powerful tool for mathematical thought.

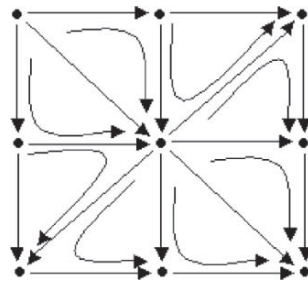
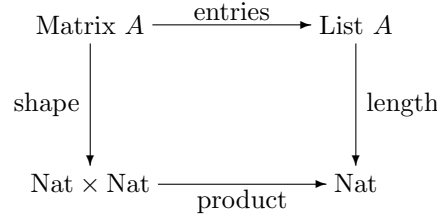


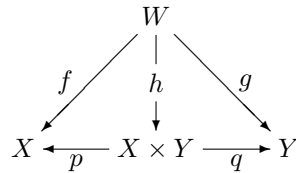
Fig. 5.1. A *commutative flow* (denoted by curved arrows) on a *triangulated digraph*. Commutativity of the whole diagram follows from commutativity of its triangular components.

As an example from linear algebra, consider an elementary diagrammatic description of matrices, using the following *pull-back diagram* [Bar93]:



asserts that a matrix is determined by its shape, given by a pair of natural numbers representing the number of rows and columns, and its data, given by the matrix entries listed in some specified order.

Many properties of mathematical constructions may be represented by *universal properties* of diagrams [MacL71]. Consider the *Cartesian product* $X \times Y$ of two sets, consisting as usual of all ordered pairs $\langle x, y \rangle$ of elements $x \in X$ and $y \in Y$. The projections $\langle x, y \rangle \mapsto x$, $\langle x, y \rangle \mapsto y$ of the product on its ‘axes’ X and Y are functions $p : X \times Y \rightarrow X$, $q : X \times Y \rightarrow Y$. Any function $h : W \rightarrow X \times Y$ from a third set W is uniquely determined by its composites $p \circ h$ and $q \circ h$. Conversely, given W and two functions f and g as in the diagram below, there is a unique function h which makes the following diagram commute:



This property describes the Cartesian product $X \times Y$ uniquely; the same diagram, read in the category of topological spaces or of groups, describes uniquely the Cartesian product of spaces or of the direct product of groups.

The construction ‘Cartesian product’ is technically called a ‘functor’ because it applies suitably both to the sets and to the functions between them; two functions $k : X \rightarrow X'$ and $l : Y \rightarrow Y'$ have a function $k \times l$ as their Cartesian product:

$$k \times l : X \times Y \rightarrow X' \times Y', \quad \langle x, y \rangle \mapsto \langle kx, ly \rangle.$$

Groups and Related Algebraic Structures

As already stated, the basic functional unit of lower biomechanics is the special Euclidean group $SE(3)$ of rigid body motions. In general, a *group* is a pointed set (G, e) with a *multiplication* $\mu : G \times G \rightarrow G$ and an *inverse* $\nu : G \rightarrow G$ such that the following diagrams commute [Swi75]:

1.

$$\begin{array}{ccccc}
 G & \xrightarrow{(e, 1)} & G \times G & \xrightarrow{(1, e)} & G \\
 & \searrow 1 & \downarrow \mu & \nearrow 1 & \\
 & & G & &
 \end{array}$$

(e is a two-sided identity)

2.

$$\begin{array}{ccc}
 G \times G \times G & \xrightarrow{\mu \times 1} & G \times G \\
 1 \times \mu \downarrow & & \downarrow \mu \\
 G \times G & \xrightarrow{\mu} & G
 \end{array}$$

(associativity)

3.

$$\begin{array}{ccccc}
 G & \xrightarrow{(\nu, 1)} & G \times G & \xrightarrow{(1, \nu)} & G \\
 & \searrow e & \downarrow \mu & \nearrow e & \\
 & & G & &
 \end{array}$$

(inverse).

Here $e : G \rightarrow G$ is the constant map $e(g) = e$ for all $g \in G$. $(e, 1)$ means the map such that $(e, 1)(g) = (e, g)$, etc. A group G is called *commutative* or *Abelian group* if in addition the following diagram commutes

$$\begin{array}{ccc}
 G \times G & \xrightarrow{T} & G \times G \\
 & \searrow \mu & \nearrow \mu \\
 & & G
 \end{array}$$

where $T : G \times G \rightarrow G \times G$ is the switch map $T(g_1, g_2) = (g_2, g_1)$, for all $(g_1, g_2) \in G \times G$.

A group G *acts* (on the left) on a set A if there is a function $\alpha : G \times A \rightarrow A$ such that the following diagrams commute [Swi75]:

1.

$$\begin{array}{ccc}
 A & \xrightarrow{(e, 1)} & G \times A \\
 & \searrow 1 & \downarrow \alpha \\
 & & A
 \end{array}$$

2.

$$\begin{array}{ccc}
 G \times G \times A & \xrightarrow{1 \times \alpha} & G \times A \\
 \mu \times 1 \downarrow & & \downarrow \alpha \\
 G \times A & \xrightarrow{\alpha} & A
 \end{array}$$

where $(e, 1)(x) = (e, x)$ for all $x \in A$. The *orbits* of the action are the sets $Gx = \{gx : g \in G\}$ for all $x \in A$.

Given two groups $(G, *)$ and $(H, \cdot)$, a *group homomorphism* from $(G, *)$ to $(H, \cdot)$ is a function $h : G \rightarrow H$ such that for all x and y in G it holds that

$$h(x * y) = h(x) \cdot h(y).$$

From this property, one can deduce that h maps the identity element e_G of G to the identity element e_H of H , and it also maps inverses to inverses in the sense that $h(x^{-1}) = h(x)^{-1}$. Hence one can say that h is *compatible* with the *group structure*.

The *kernel* $\text{Ker } h$ of a group homomorphism $h : G \rightarrow H$ consists of all those elements of G which are sent by h to the identity element e_H of H , i.e.,

$$\text{Ker } h = \{x \in G : h(x) = e_H\}.$$

The *image* $\text{Im } h$ of a group homomorphism $h : G \rightarrow H$ consists of all elements of H which are sent by h to H , i.e.,

$$\text{Im } h = \{h(x) : x \in G\}.$$

The kernel is a *normal subgroup* of G and the image is a *subgroup* of H . The homomorphism h is *injective* (and called a *group monomorphism*) iff $\text{Ker } h = e_G$, i.e., iff the kernel of h consists of the identity element of G only.

Similarly, a *ring* is a set S together with two binary operators $+$ and $*$ (commonly interpreted as addition and multiplication, respectively) satisfying the following conditions:

1. Additive associativity: For all $a, b, c \in S$, $(a + b) + c = a + (b + c)$,
2. Additive commutativity: For all $a, b \in S$, $a + b = b + a$,
3. Additive identity: There exists an element $0 \in S$ such that for all $a \in S$, $0 + a = a + 0 = a$,
4. Additive inverse: For every $a \in S$ there exists $-a \in S$ such that $a + (-a) = (-a) + a = 0$,
5. Multiplicative associativity: For all $a, b, c \in S$, $(a * b) * c = a * (b * c)$,
6. Left and right distributivity: For all $a, b, c \in S$, $a * (b + c) = (a * b) + (a * c)$ and $(b + c) * a = (b * a) + (c * a)$.

A ring (the term introduced by *David Hilbert*) is therefore an Abelian group under addition and a semigroup under multiplication. A ring that is commutative under multiplication, has a unit element, and has no divisors of zero is called an *integral domain*. A ring which is also a commutative multiplication group is called a *field*. The simplest rings are the integers $\mathbb{Z}$, polynomials $R[x]$ and $R[x, y]$ in one and two variables, and square $n \times n$ real matrices.

An *ideal* is a subset $\mathfrak{I}$ of elements in a ring R which forms an additive group and has the property that, whenever x belongs to R and y belongs to $\mathfrak{I}$, then xy and yx belong to $\mathfrak{I}$. For example, the set of even integers is an ideal in the ring of integers $\mathbb{Z}$. Given an ideal $\mathfrak{I}$, it is possible to define a factor ring $R/\mathfrak{I}$.

A ring is called *left* (respectively, *right*) *Noetherian* if it does not contain an infinite ascending chain of left (respectively, right) ideals. In this case, the ring in question is said to satisfy the ascending chain condition on left (respectively, right) ideals. A *ring* is said to be *Noetherian* if it is both left and right Noetherian. If a ring R is Noetherian, then the following are equivalent:

1. R satisfies the ascending chain condition on ideals.
2. Every ideal of R is finitely generated.
3. Every set of ideals contains a maximal element.

A *module* is a mathematical object in which things can be added together commutatively by multiplying coefficients and in which most of the rules of manipulating vectors hold. A module is abstractly very similar to a vector space, although in modules, coefficients are taken in rings which are much more general algebraic objects than the fields used in vector spaces. A module taking its coefficients in a ring R is called a module over R or R -module. Modules are the basic tool of homological algebra.

Examples of modules include the set of integers $\mathbb{Z}$, the cubic lattice in d dimensions $\mathbb{Z}^d$, and the group ring of a group. $\mathbb{Z}$ is a module over itself. It is closed under addition and subtraction. Numbers of the form $n\alpha$ for $n \in \mathbb{Z}$ and α a fixed integer form a submodule since, for $(n, m) \in \mathbb{Z}$, $n\alpha \pm m\alpha = (n \pm m)\alpha$ and $(n \pm m)$ is still in $\mathbb{Z}$. Also, given two integers a and b , the smallest module containing a and b is the module for their greatest common divisor, $\alpha = \text{GCD}(a, b)$.

A module M is a *Noetherian module* if it obeys the ascending chain condition with respect to inclusion, i.e., if every set of increasing sequences of submodules eventually becomes constant. If a module M is Noetherian, then the following are equivalent:

1. M satisfies the ascending chain condition on submodules.
2. Every submodule of M is finitely generated.
3. Every set of submodules of M contains a maximal element.

Let I be a partially ordered set. A *direct system* of R -modules over I is an ordered pair $\{M_i, \varphi_j^i\}$ consisting of an indexed family of modules $\{M_i : i \in I\}$

together with a family of homomorphisms $\{\varphi_j^i : M_i \rightarrow M_j\}$ for $i \leq j$, such that $\varphi_i^i = 1_{M_i}$ for all i and such that the following diagram commutes whenever $i \leq j \leq k$

$$\begin{array}{ccc} M_i & \xrightarrow{\varphi_k^i} & M_k \\ & \searrow \varphi_k^j & \nearrow \varphi_j^i \\ & M_j & \end{array}$$

Similarly, an *inverse system* of R -modules over I is an ordered pair $\{M_i, \psi_i^j\}$ consisting of an indexed family of modules $\{M_i : i \in I\}$ together with a family of homomorphisms $\{\psi_i^j : M_j \rightarrow M_i\}$ for $i \leq j$, such that $\psi_i^i = 1_{M_i}$ for all i and such that the following diagram commutes whenever $i \leq j \leq k$

$$\begin{array}{ccc} M_k & \xrightarrow{\psi_i^k} & M_i \\ & \searrow \psi_j^k & \nearrow \psi_i^j \\ & M_j & \end{array}$$

5.1.2 Categories

A category is a generic mathematical structure consisting of a collection of *objects* (sets with possibly additional structure), with a corresponding collection of *arrows*, or *morphisms*, between objects (agreeing with this additional structure). A category $\mathcal{K}$ is defined as a pair $(\mathbf{Ob}(\mathcal{K}), \mathbf{Mor}(\mathcal{K}))$ of generic objects $A, B, \dots$ in $\mathbf{Ob}(\mathcal{K})$ and generic arrows $f : A \rightarrow B, g : B \rightarrow C, \dots$ in $\mathbf{Mor}(\mathcal{K})$ between objects, with *associative composition*:

$$A \xrightarrow{f} B \xrightarrow{g} C = A \xrightarrow{f \circ g} C,$$

and *identity (loop) arrow*. (Note that in topological literature, $\mathbf{Hom}(\mathcal{K})$ or $\mathbf{hom}(\mathcal{K})$ is used instead of $\mathbf{Mor}(\mathcal{K})$; see [Swi75]).

A category $\mathcal{K}$ is usually depicted as a *commutative diagram* (i.e., a diagram with a common *initial object* A and *final object* D):

$$\begin{array}{ccc} A & \xrightarrow{f} & B \\ h \downarrow & \mathcal{K} & \downarrow g \\ C & \xrightarrow{k} & D \end{array}$$

To make this more precise, we say that a *category* $\mathcal{K}$ is defined if we have:

1. A *class of objects* $\{A, B, C, \dots\}$ of $\mathcal{K}$, denoted by $\mathbf{Ob}(\mathcal{K})$;
2. A *set of morphisms*, or *arrows* $\mathbf{Mor}_{\mathcal{K}}(A, B)$, with elements $f : A \rightarrow B$, defined for any *ordered pair* $(A, B) \in \mathcal{K}$, such that for two different pairs $(A, B) \neq (C, D)$ in $\mathcal{K}$, we have $\mathbf{Mor}_{\mathcal{K}}(A, B) \cap \mathbf{Mor}_{\mathcal{K}}(C, D) = \emptyset$;
3. For any *triplet* $(A, B, C) \in \mathcal{K}$ with $f : A \rightarrow B$ and $g : B \rightarrow C$, there is a *composition* of morphisms

$$\mathbf{Mor}_{\mathcal{K}}(B, C) \times \mathbf{Mor}_{\mathcal{K}}(A, B) \ni (g, f) \rightarrow g \circ f \in \mathbf{Mor}_{\mathcal{K}}(A, C),$$

written schematically as

$$\frac{f : A \rightarrow B, \quad g : B \rightarrow C}{g \circ f : A \rightarrow C}.$$

If we have a morphism $f \in \mathbf{Mor}_{\mathcal{K}}(A, B)$, (otherwise written $f : A \rightarrow B$, or $A \xrightarrow{f} B$), then $A = \text{dom}(f)$ is a *domain* of f , and $B = \text{cod}(f)$ is a *codomain* of f (of which *range* of f is a subset) and denoted $B = \text{ran}(f)$.

To make $\mathcal{K}$ a category, it must also fulfill the following two properties:

1. *Associativity of morphisms*: for all $f \in \mathbf{Mor}_{\mathcal{K}}(A, B)$, $g \in \mathbf{Mor}_{\mathcal{K}}(B, C)$, and $h \in \mathbf{Mor}_{\mathcal{K}}(C, D)$, we have $h \circ (g \circ f) = (h \circ g) \circ f$; in other words, the following diagram is commutative

$$\begin{array}{ccc} A & \xrightarrow{h \circ (g \circ f) = (h \circ g) \circ f} & D \\ f \downarrow & & \uparrow h \\ B & \xrightarrow{g} & C \end{array}$$

2. *Existence of identity morphism*: for every object $A \in \mathbf{Ob}(\mathcal{K})$ exists a unique identity morphism $1_A \in \mathbf{Mor}_{\mathcal{K}}(A, A)$; for any two morphisms $f \in \mathbf{Mor}_{\mathcal{K}}(A, B)$, and $g \in \mathbf{Mor}_{\mathcal{K}}(B, C)$, compositions with identity morphism $1_B \in \mathbf{Mor}_{\mathcal{K}}(B, B)$ give $1_B \circ f = f$ and $g \circ 1_B = g$, i.e., the following diagram is commutative:

$$\begin{array}{ccccc} A & \xrightarrow{f} & B & \xrightarrow{g} & C \\ & \searrow f & \downarrow 1_B & \nearrow g & \\ & & B & & \end{array}$$

The set of all morphisms of the category $\mathcal{K}$ is denoted

$$\mathbf{Mor}(\mathcal{K}) = \bigcup_{A, B \in \mathbf{Ob}(\mathcal{K})} \mathbf{Mor}_{\mathcal{K}}(A, B).$$

If for two morphisms $f \in \text{Mor}_{\mathcal{K}}(A, B)$ and $g \in \text{Mor}_{\mathcal{K}}(B, A)$ the equality $g \circ f = 1_A$ is valid, then the morphism g is said to be *left inverse* (or *retraction*), of f , and f *right inverse* (or *section*) of g . A morphism which is both right and left inverse of f is said to be *two-sided inverse* of f .

A morphism $m : A \rightarrow B$ is called *monomorphism* in $\mathcal{K}$ (i.e., *1-1*, or *injection* map), if for any two parallel morphisms $f_1, f_2 : C \rightarrow A$ in $\mathcal{K}$ the equality $m \circ f_1 = m \circ f_2$ implies $f_1 = f_2$; in other words, m is monomorphism if it is *left cancelable*. Any morphism with a left inverse is monomorphism.

A morphism $e : A \rightarrow B$ is called *epimorphism* in $\mathcal{K}$ (i.e., *onto*, or *surjection* map), if for any two morphisms $g_1, g_2 : B \rightarrow C$ in $\mathcal{K}$ the equality $g_1 \circ e = g_2 \circ e$ implies $g_1 = g_2$; in other words, e is epimorphism if it is *right cancelable*. Any morphism with a right inverse is epimorphism.

A morphism $f : A \rightarrow B$ is called *isomorphism* in $\mathcal{K}$ (denoted as $f : A \cong B$) if there exists a morphism $f^{-1} : B \rightarrow A$ which is a two-sided inverse of f in $\mathcal{K}$. The relation of isomorphism is reflexive, symmetric, and transitive, i.e., equivalence relation.

For example, an isomorphism in the category of sets is called a set-isomorphism, or a *bijection*, in the category of topological spaces is called a topological isomorphism, or a *homeomorphism*, in the category of differentiable manifolds is called a differentiable isomorphism, or a *diffeomorphism*.

A morphism $f \in \text{Mor}_{\mathcal{K}}(A, B)$ is *regular* if there exists a morphism $g : B \rightarrow A$ in $\mathcal{K}$ such that $f \circ g \circ f = f$. Any morphism with either a left or a right inverse is regular.

An object T is a *terminal object* in $\mathcal{K}$ if to each object $A \in \text{Ob}(\mathcal{K})$ there is exactly one arrow $A \rightarrow T$. An object S is an *initial object* in $\mathcal{K}$ if to each object $A \in \text{Ob}(\mathcal{K})$ there is exactly one arrow $S \rightarrow A$. A *null object* $Z \in \text{Ob}(\mathcal{K})$ is an object which is both initial and terminal; it is unique up to isomorphism. For any two objects $A, B \in \text{Ob}(\mathcal{K})$ there is a unique morphism $A \rightarrow Z \rightarrow B$ (the composite through Z), called the *zero morphism* from A to B .

A notion of subcategory is analogous to the notion of subset. A subcategory $\mathcal{L}$ of a category $\mathcal{K}$ is said to be a *complete subcategory* iff for any objects $A, B \in \mathcal{L}$, every morphism $A \rightarrow B$ of $\mathcal{L}$ is in $\mathcal{K}$.

A *groupoid* is a category in which every morphism is invertible. A typical groupoid is the *fundamental groupoid* $\Pi_1(X)$ of a topological space X . An object of $\Pi_1(X)$ is a point $x \in X$, and a morphism $x \rightarrow x'$ of $\Pi_1(X)$ is a homotopy class of paths f from x to x' . The *composition* of paths $g : x' \rightarrow x''$ and $f : x \rightarrow x'$ is the path h which is ‘ f followed by g ’. Composition applies also to homotopy classes, and makes $\Pi_1(X)$ a category and a groupoid (the inverse of any path is the same path traced in the opposite direction).

A *group* is a groupoid with one object, i.e., a *category with one object* in which *all morphisms are isomorphisms*. Therefore, if we try to generalize the concept of a group, keeping associativity as an essential property, we get the notion of a category.

A category is *discrete* if every morphism is an identity. A *monoid* is a category with one object. A *group* is a category with one object in which every morphism has a two-sided inverse under composition.

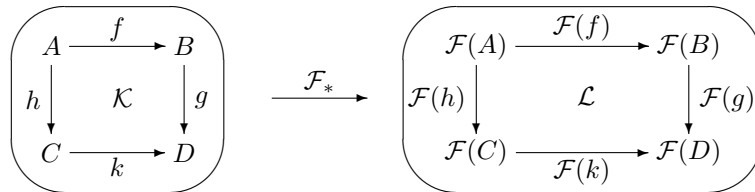
Homological algebra was the progenitor of category theory (see e.g., [Die88]). Generalizing L. Euler's formula $f + v = e + 2$ for the faces, vertices and edges of a convex polyhedron, E. Betti defined *numerical invariants of spaces* by formal addition and subtraction of faces of various dimensions; H. Poincaré formalized these and introduced homology. E. Noether stressed the fact that these calculations go on in Abelian groups, and that the operation ∂_n taking a face of dimension n to the alternating sum of faces of dimension $n - 1$ which form its boundary is a homomorphism, and it also satisfies $\partial_n \circ \partial_{n+1} = 0$. There are many ways of approximating a given space by polyhedra, but the quotient $H_n = \text{Ker } \partial_n / \text{Im } \partial_{n+1}$ is an invariant, the *homology group*. Since Noether, the groups have been the object of study instead of their dimensions, which are the *Betti numbers*.

5.1.3 Functors

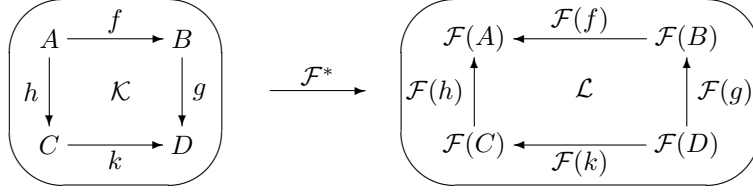
In algebraic topology, one attempts to assign to every topological space X some algebraic object $\mathcal{F}(X)$ in such a way that to every C^0 -function $f : X \rightarrow Y$ there is assigned a homomorphism $\mathcal{F}(f) : \mathcal{F}(X) \rightarrow \mathcal{F}(Y)$ (see [Swi75, II06b]). One advantage of this procedure is, e.g., that if one is trying to prove the non-existence of a C^0 -function $f : X \rightarrow Y$ with certain properties, one may find it relatively easy to prove the non-existence of the corresponding algebraic function $\mathcal{F}(f)$ and hence deduce that f could not exist. In other words, $\mathcal{F}$ is to be a 'homomorphism' from one category (e.g., $\mathcal{T}$) to another (e.g., $\mathcal{G}$ or $\mathcal{A}$). Formalization of this notion is a *functor*.

A functor is a generic *picture* projecting (all objects and morphisms of) a source category into a target category. Let $\mathcal{K} = (\text{Ob}(\mathcal{K}), \text{Mor}(\mathcal{K}))$ be a *source* (or domain) *category* and $\mathcal{L} = (\text{Ob}(\mathcal{L}), \text{Mor}(\mathcal{L}))$ be a *target* (or codomain) category. A functor $\mathcal{F} = (\mathcal{F}_O, \mathcal{F}_M)$ is defined as a pair of maps, $\mathcal{F}_O : \text{Ob}(\mathcal{K}) \rightarrow \text{Ob}(\mathcal{L})$ and $\mathcal{F}_M : \text{Mor}(\mathcal{K}) \rightarrow \text{Mor}(\mathcal{L})$, preserving categorical symmetry (i.e., commutativity of all diagrams) of $\mathcal{K}$ in $\mathcal{L}$.

More precisely, a *covariant functor*, or simply a *functor*, $\mathcal{F}_* : \mathcal{K} \rightarrow \mathcal{L}$ is a *picture* in the target category $\mathcal{L}$ of (all objects and morphisms of) the source category $\mathcal{K}$:



Similarly, a *contravariant functor*, or a *cofunctor*, $\mathcal{F}^* : \mathcal{K} \rightarrow \mathcal{L}$ is a *dual picture* with reversed arrows:



In other words, a *functor* $\mathcal{F} : \mathcal{K} \rightarrow \mathcal{L}$ from a *source* category $\mathcal{K}$ to a *target* category $\mathcal{L}$, is a pair $\mathcal{F} = (\mathcal{F}_O, \mathcal{F}_M)$ of maps $\mathcal{F}_O : \mathbf{Ob}(\mathcal{K}) \rightarrow \mathbf{Ob}(\mathcal{L})$, $\mathcal{F}_M : \mathbf{Mor}(\mathcal{K}) \rightarrow \mathbf{Mor}(\mathcal{L})$, such that

1. If $f \in \mathbf{Mor}_{\mathcal{K}}(A, B)$ then $\mathcal{F}_M(f) \in \mathbf{Mor}_{\mathcal{L}}(\mathcal{F}_O(A), \mathcal{F}_O(B))$ in case of the *covariant* functor $\mathcal{F}_*$, and $\mathcal{F}_M(f) \in \mathbf{Mor}_{\mathcal{L}}(\mathcal{F}_O(B), \mathcal{F}_O(A))$ in case of the *contravariant* functor $\mathcal{F}^*$;
2. For all $A \in \mathbf{Ob}(\mathcal{K})$: $\mathcal{F}_M(1_A) = 1_{\mathcal{F}_O(A)}$;
3. For all $f, g \in \mathbf{Mor}(\mathcal{K})$: if $\text{cod}(f) = \text{dom}(g)$, then $\mathcal{F}_M(g \circ f) = \mathcal{F}_M(g) \circ \mathcal{F}_M(f)$ in case of the *covariant* functor $\mathcal{F}_*$, and $\mathcal{F}_M(g \circ f) = \mathcal{F}_M(f) \circ \mathcal{F}_M(g)$ in case of the *contravariant* functor $\mathcal{F}^*$.

Category theory originated in algebraic topology, which tried to assign algebraic invariants to topological structures. The golden rule of such *invariants* is that they should be *functors*. For example, the *fundamental group* π_1 is a functor. Algebraic topology constructs a group called the *fundamental group* $\pi_1(X)$ from any topological space X , which keeps track of how many holes the space X has. But also, any map between topological spaces determines a homomorphism $\phi : \pi_1(X) \rightarrow \pi_1(Y)$ of the fundamental groups. So the fundamental group is really a functor $\pi_1 : \mathcal{T} \rightarrow \mathcal{G}$. This allows us to completely transpose any situation involving *spaces* and *continuous maps* between them to a parallel situation involving *groups* and *homomorphisms* between them, and thus reduce some topology problems to algebra problems.

Also, singular homology in a given dimension n assigns to each topological space X an Abelian group $H_n(X)$, its *nth homology group* of X , and also to each continuous map $f : X \rightarrow Y$ of spaces a corresponding homomorphism $H_n(f) : H_n(X) \rightarrow H_n(Y)$ of groups, and this in such a way that $H_n(X)$ becomes a functor $H_n : \mathcal{T} \rightarrow \mathcal{A}$.

The leading idea in the *use of functors in topology* is that H_n or π_n gives an algebraic picture or image not just of the topological spaces X, Y but also of all the continuous maps $f : X \rightarrow Y$ between them.

Similarly, there is a functor $\Pi_1 : \mathcal{T} \rightarrow \mathcal{G}$, called the ‘fundamental groupoid functor’, which plays a very basic role in algebraic topology. Here’s how we get from any space X its ‘fundamental groupoid’ $\Pi_1(X)$. To say what the groupoid $\Pi_1(X)$ is, we need to say what its objects and morphisms are. The objects in $\Pi_1(X)$ are just the *points* of X and the morphisms are just certain equivalence classes of *paths* in X . More precisely, a morphism $f : x \rightarrow y$ in $\Pi_1(X)$ is just an equivalence class of continuous paths from x to y , where two paths from x to y are decreed equivalent if one can be continuously deformed to

the other while not moving the endpoints. (If this equivalence relation holds we say the two paths are ‘homotopic’, and we call the equivalence classes ‘homotopy classes of paths’ (see [MacL71, Swi75]).

Another examples are covariant *forgetful* functors:

- From the category of topological spaces to the category of sets; it ‘forgets’ the topology–structure.
- From the category of metric spaces to the category of topological spaces with the topology induced by the metrics; it ‘forgets’ the metric.

For each category $\mathcal{K}$, the *identity functor* $I_{\mathcal{K}}$ takes every $\mathcal{K}$ –object and every $\mathcal{K}$ –morphism to itself.

Given a category $\mathcal{K}$ and its subcategory $\mathcal{L}$, we have an *inclusion functor* $\text{In} : \mathcal{K} \rightarrow \mathcal{K}$.

Given a category $\mathcal{K}$, a *diagonal functor* $\Delta : \mathcal{K} \rightarrow \mathcal{K}$ takes each object $A \in \mathcal{K}$ to the object (A, A) in the product category $\mathcal{K} \times \mathcal{K}$.

Given a category $\mathcal{K}$ and a category of sets $\mathcal{S}$, each object $A \in \mathcal{K}$ determines a *covariant Hom–functor* $\mathcal{K}[A, \_] : \mathcal{K} \rightarrow \mathcal{S}$, a *contravariant Hom–functor* $\mathcal{K}[\_, A] : \mathcal{K} \rightarrow \mathcal{S}$, and a *Hom–bifunctor* $\mathcal{K}[\_, \_] : \mathcal{K}^{op} \times \mathcal{K} \rightarrow \mathcal{S}$.

A functor $\mathcal{F} : \mathcal{K} \rightarrow \mathcal{L}$ is a *faithful functor* if for all $A, B \in \text{Ob}(\mathcal{K})$ and for all $f, g \in \text{Mor}_{\mathcal{K}}(A, B)$, $\mathcal{F}(f) = \mathcal{F}(g)$ implies $f = g$; it is a *full functor* if for every $h \in \text{Mor}_{\mathcal{L}}(\mathcal{F}(A), \mathcal{F}(B))$, there is $g \in \text{Mor}_{\mathcal{K}}(A, B)$ such that $h = \mathcal{F}(g)$; it is a *full embedding* if it is both full and faithful.

A *representation of a group* is a functor $\mathcal{F} : \mathcal{G} \rightarrow \mathcal{V}$.

Similarly, we can define a *representation of a category* to be a functor $\mathcal{F} : \mathcal{K} \rightarrow \mathcal{V}$ from the 2–category $\mathcal{K}$ (a ‘big’ category including all ordinary, or ‘small’ categories, see subsection (5.1.7) below) to the category of vector spaces $\mathcal{V}$. In this way, a category is a generalization of a group and group representations are a special case of category representations.

5.1.4 Natural Transformations

A *natural transformation* (i.e., a *functor morphism*) $\tau : \mathcal{F} \rightarrow \mathcal{G}$ is a *map between two functors of the same variance*, $(\mathcal{F}, \mathcal{G}) : \mathcal{K} \rightrightarrows \mathcal{L}$, preserving categorical symmetry:

$$\begin{array}{ccc}
 \boxed{A \xrightarrow{f} B} & \xrightarrow{\mathcal{F}} & \boxed{\begin{array}{ccc} \mathcal{F}(A) & \xrightarrow{\mathcal{F}(f)} & \mathcal{F}(B) \\ \tau_A \downarrow & \mathcal{L} & \downarrow \tau_B \\ \mathcal{G}(A) & \xrightarrow{\mathcal{G}(f)} & \mathcal{G}(B) \end{array}} \\
 \mathcal{K} & \xrightarrow{\mathcal{G}} &
 \end{array}$$

More precisely, all functors of the same variance from a source category $\mathcal{K}$ to a target category $\mathcal{L}$ form themselves objects of the *functor category* $\mathcal{L}^{\mathcal{K}}$. Morphisms of $\mathcal{L}^{\mathcal{K}}$, called *natural transformations*, are defined as follows.

Let $\mathcal{F} : \mathcal{K} \rightarrow \mathcal{L}$ and $\mathcal{G} : \mathcal{K} \rightarrow \mathcal{L}$ be two functors of the same variance from a category $\mathcal{K}$ to a category $\mathcal{L}$. Natural transformation $\mathcal{F} \xrightarrow{\tau} \mathcal{G}$ is a family of morphisms such that for all $f \in \text{Mor}_{\mathcal{K}}(A, B)$ in the source category $\mathcal{K}$, we have $\mathcal{G}(f) \circ \tau_A = \tau_B \circ \mathcal{F}(f)$ in the target category $\mathcal{L}$. Then we say that the component $\tau_A : \mathcal{F}(A) \rightarrow \mathcal{G}(A)$ is natural in A .

If we think of a functor $\mathcal{F}$ as giving a *picture* in the target category $\mathcal{L}$ of (all the objects and morphisms of) the source category $\mathcal{K}$, then a natural transformation τ represents a set of morphisms mapping the picture $\mathcal{F}$ to another picture $\mathcal{G}$, preserving the commutativity of all diagrams.

An invertible natural transformation, such that all components τ_A are isomorphisms) is called a *natural equivalence* (or, *natural isomorphism*). In this case, the inverses $(\tau_A)^{-1}$ in $\mathcal{L}$ are the components of a natural isomorphism $(\tau)^{-1} : \mathcal{G} \xrightarrow{*} \mathcal{F}$. Natural equivalences are among the most important *metamathematical constructions* in algebraic topology (see [Swi75]).

For example, let $\mathcal{B}$ be the category of Banach spaces over $\mathbb{R}$ and bounded linear maps. Define $D : \mathcal{B} \rightarrow \mathcal{B}$ by taking $D(X) = X^* =$ Banach space of bounded linear functionals on a space X and $D(f) = f^*$ for $f : X \rightarrow Y$ a bounded linear map. Then D is a cofunctor. $D^2 = D \circ D$ is also a functor. We also have the identity functor $1 : \mathcal{B} \rightarrow \mathcal{B}$. Define $T : 1 \rightarrow D \circ D$ as follows: for every $X \in \mathcal{B}$ let $T(X) : X \rightarrow D^2X = X^{**}$ be the *natural inclusion* – that is, for $x \in X$ we have $[T(X)(x)](f) = f(x)$ for every $f \in X^*$. T is a natural transformation. On the subcategory of n D Banach spaces T is even a natural equivalence. The largest subcategory of $\mathcal{B}$ on which T is a natural equivalence is called the category of reflexive Banach spaces [Swi75].

As S. Eilenberg and S. MacLane first observed, ‘category’ has been defined in order to define ‘functor’ and ‘functor’ has been defined in order to define ‘natural transformation’ [MacL71]).

Compositions of Natural Transformations

Natural transformations can be *composed* in two different ways. First, we have an ‘ordinary’ composition: if $\mathcal{F}, \mathcal{G}$ and $\mathcal{H}$ are three functors from the source category $\mathcal{A}$ to the target category $\mathcal{B}$, and then $\alpha : \mathcal{F} \rightarrow \mathcal{G}$, $\beta : \mathcal{G} \rightarrow \mathcal{H}$ are two natural transformations, then the formula

$$(\beta \circ \alpha)_A = \beta_A \circ \alpha_A, \quad \text{for all } A \in \mathcal{A}, \quad (5.2)$$

defines a new natural transformation $\beta \circ \alpha : \mathcal{F} \rightarrow \mathcal{H}$. This composition law is clearly associative and possesses a unit $1_{\mathcal{F}}$ at each functor $\mathcal{F}$, whose $\mathcal{A}$ -component is $1_{\mathcal{F}A}$.

Second, we have the *Godement product* of natural transformations, usually denoted by $*$. Let $\mathcal{A}, \mathcal{B}$ and $\mathcal{C}$ be three categories, $\mathcal{F}, \mathcal{G}, \mathcal{H}$ and $\mathcal{K}$ be four functors such that $(\mathcal{F}, \mathcal{G}) : \mathcal{A} \rightrightarrows \mathcal{B}$ and $(\mathcal{H}, \mathcal{K}) : \mathcal{B} \rightrightarrows \mathcal{C}$, and $\alpha : \mathcal{F} \rightarrow \mathcal{G}$, $\beta : \mathcal{H} \rightarrow \mathcal{K}$ be two natural transformations. Now, instead of (5.2), the Godement composition is given by

$$(\beta * \alpha)_A = \beta_{GA} \circ H(\alpha_A) = K(\alpha_A) \circ \beta_{FA}, \quad \text{for all } A \in \mathcal{A}, \quad (5.3)$$

which defines a new natural transformation $\beta * \alpha : \mathcal{H} \circ \mathcal{F} \rightarrow \mathcal{K} \circ \mathcal{G}$.

Finally, the two compositions (5.2) and (5.2) of natural transformations can be combined as

$$(\delta * \gamma) \circ (\beta * \alpha) = (\delta \circ \beta) * (\gamma \circ \alpha),$$

where $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$ are three categories, $\mathcal{F}, \mathcal{G}, \mathcal{H}, \mathcal{K}, \mathcal{L}, \mathcal{M}$ are six functors, and $\alpha : \mathcal{F} \rightarrow \mathcal{H}$, $\beta : \mathcal{G} \rightarrow \mathcal{K}$, $\gamma : \mathcal{H} \rightarrow \mathcal{L}$, $\delta : \mathcal{K} \rightarrow \mathcal{M}$ are four natural transformations.

Dinatural Transformations

Double natural transformations are called *dinatural transformations*. An *end of a functor* $S : C^{op} \times C \rightarrow X$ is a universal dinatural transformation from a constant e to S . In other words, an end of S is a pair $\langle e, \omega \rangle$, where e is an object of X and $\omega : e \rightarrow S$ is a *wedge (dinatural) transformation* with the property that to every wedge $\beta : x \rightarrow S$ there is a unique arrow $h : x \rightarrow e$ of B with $\beta_c = \omega_c h$ for all $a \in C$. We call ω the *ending wedge with components* ω_c , while the object e itself, by abuse of language, is called the end of S and written with integral notation as $\int_c S(c, c)$; thus

$$S(c, c) \xrightarrow{\omega_c} \int_c S(c, c) = e.$$

Note that the ‘variable of integration’ c appears twice under the integral sign (once contravariant, once covariant) and is ‘bound’ by the integral sign, in that the result no longer depends on c and so is unchanged if ‘ c ’ is replaced by any other letter standing for an object of the category C . These properties are like those of the letter x under the usual integral symbol $\int f(x) dx$ of calculus.

Every end is manifestly a limit (see below) – specifically, a limit of a suitable diagram in X made up of pieces like $S(b, b) \rightarrow S(b, c) \rightarrow S(c, c)$.

For each functor $T : C \rightarrow X$ there is an isomorphism

$$\int_c S(c, c) = \int_c Tc \cong \text{Lim } T,$$

valid when either the end of the limit exists, carrying the ending wedge to the limiting cone; the indicated notation thus allows us to write any limit as an integral (an end) without explicitly mentioning the dummy variable (the first variable c of S).

A functor $H : X \rightarrow Y$ is said to *preserve the end* of a functor $S : C^{op} \times C \rightarrow X$ when $\omega : e \rightarrow S$ an end of S in X implies that $H\omega : He \rightarrow HS$ is an end for HS ; in symbols

$$H \int_c S(c, c) = \int_c HS(c, c).$$

Similarly, H creates the end of S when to each end $v : y \rightrightarrows HS$ in Y there is a unique wedge $\omega : e \rightrightarrows S$ with $H\omega = v$, and this wedge ω is an end of S .

The definition of the coend of a functor $S : C^{op} \times C \rightarrow X$ is dual to that of an end. A *coend* of S is a pair $\langle d, \zeta \rangle$, consisting of an object $d \in X$ and a wedge $\zeta : S \rightrightarrows d$. The object d (when it exists, unique up to isomorphism) will usually be written with an integral sign and with the bound variable c as superscript; thus

$$S(c, c) \xrightarrow{\zeta} \int_c S(c, c) = d.$$

The formal properties of coends are dual to those of ends. Both are much like those for integrals in calculus (see [MacL71], for technical details).

5.1.5 Limits and Colimits

In abstract algebra constructions are often defined by an abstract property which requires the existence of unique morphisms under certain conditions. These properties are called *universal properties*. The *limit* of a functor generalizes the notions of inverse limit and product used in various parts of mathematics. The dual notion, *colimit*, generalizes direct limits and direct sums. Limits and colimits are defined via universal properties and provide many examples of *adjoint functors*.

A *limit* of a covariant functor $\mathcal{F} : \mathcal{J} \rightarrow \mathcal{C}$ is an object L of $\mathcal{C}$, together with morphisms $\phi_X : L \rightarrow \mathcal{F}(X)$ for every object X of $\mathcal{J}$, such that for every morphism $f : X \rightarrow Y$ in $\mathcal{J}$, we have $\mathcal{F}(f)\phi_X = \phi_Y$, and such that the following *universal property* is satisfied: for any object N of $\mathcal{C}$ and any set of morphisms $\psi_X : N \rightarrow \mathcal{F}(X)$ such that for every morphism $f : X \rightarrow Y$ in $\mathcal{J}$, we have $\mathcal{F}(f)\psi_X = \psi_Y$, there exists precisely one morphism $u : N \rightarrow L$ such that $\phi_X u = \psi_X$ for all X . If $\mathcal{F}$ has a limit (which it need not), then the limit is defined up to a unique isomorphism, and is denoted by $\lim \mathcal{F}$.

Analogously, a *colimit* of the functor $\mathcal{F} : \mathcal{J} \rightarrow \mathcal{C}$ is an object L of $\mathcal{C}$, together with morphisms $\phi_X : \mathcal{F}(X) \rightarrow L$ for every object X of $\mathcal{J}$, such that for every morphism $f : X \rightarrow Y$ in $\mathcal{J}$, we have $\phi_Y \mathcal{F}(f) = \phi_X$, and such that the following universal property is satisfied: for any object N of $\mathcal{C}$ and any set of morphisms $\psi_X : \mathcal{F}(X) \rightarrow N$ such that for every morphism $f : X \rightarrow Y$ in $\mathcal{J}$, we have $\psi_Y \mathcal{F}(f) = \psi_X$, there exists precisely one morphism $u : L \rightarrow N$ such that $u\phi_X = \psi_X$ for all X . The colimit of $\mathcal{F}$, unique up to unique isomorphism if it exists, is denoted by $\text{colim } \mathcal{F}$.

Limits and colimits are related as follows: A functor $\mathcal{F} : \mathcal{J} \rightarrow \mathcal{C}$ has a colimit iff for every object N of $\mathcal{C}$, the functor $X \mapsto \text{Mor}_{\mathcal{C}}(\mathcal{F}(X), N)$ (which is a covariant functor on the dual category $\mathcal{J}^{op}$) has a limit. If that is the case, then $\text{Mor}_{\mathcal{C}}(\text{colim } \mathcal{F}, N) = \lim \text{Mor}_{\mathcal{C}}(\mathcal{F}(-), N)$ for every object N of $\mathcal{C}$.

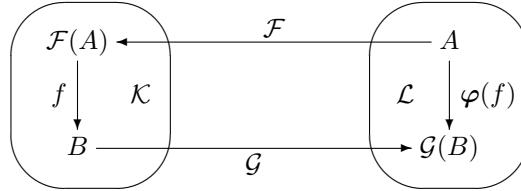
5.1.6 Adjunction

The most important functorial operation is adjunction; as S. MacLane once said, “Adjoint functors arise everywhere” [MacL71].

The *adjunction* $\varphi : \mathcal{F} \dashv \mathcal{G}$ between two functors $(\mathcal{F}, \mathcal{G}) : \mathcal{K} \rightleftharpoons \mathcal{L}$ of *opposite variance* [Kan88], represents a *weak functorial inverse*

$$\frac{f : \mathcal{F}(A) \rightarrow B}{\varphi(f) : A \rightarrow \mathcal{G}(B)}$$

forming a *natural equivalence* $\varphi : \text{Mor}_{\mathcal{K}}(\mathcal{F}(A), B) \xrightarrow{\varphi} \text{Mor}_{\mathcal{L}}(A, \mathcal{G}(B))$. The adjunction isomorphism is given by a *bijective correspondence* (a 1–1 and onto map on objects) $\varphi : \text{Mor}(\mathcal{K}) \ni f \rightarrow \varphi(f) \in \text{Mor}(\mathcal{L})$ of isomorphisms in the two categories, $\mathcal{K}$ (with a representative object A), and $\mathcal{L}$ (with a representative object B). It can be depicted as a (non-commutative) diagram



In this case $\mathcal{F}$ is called *left adjoint*, while $\mathcal{G}$ is called *right adjoint*.

In other words, an *adjunction* $F \dashv G$ between two functors $(\mathcal{F}, \mathcal{G})$ of opposite variance, from a source category $\mathcal{K}$ to a target category $\mathcal{L}$, is denoted by $(\mathcal{F}, \mathcal{G}, \eta, \epsilon) : \mathcal{K} \rightleftharpoons \mathcal{L}$. Here, $\mathcal{F} : \mathcal{L} \rightarrow \mathcal{K}$ is the *left (upper) adjoint functor*, $\mathcal{G} : \mathcal{K} \leftarrow \mathcal{L}$ is the *right (lower) adjoint functor*, $\eta : 1_{\mathcal{L}} \rightarrow \mathcal{G} \circ \mathcal{F}$ is the *unit natural transformation* (or, *front adjunction*), and $\epsilon : \mathcal{F} \circ \mathcal{G} \rightarrow 1_{\mathcal{K}}$ is the *counit natural transformation* (or, *back adjunction*).

For example, $\mathcal{K} = \mathcal{S}$ is the category of sets and $\mathcal{L} = \mathcal{G}$ is the category of groups. Then $\mathcal{F}$ turns any set into the *free group* on that set, while the ‘forgetful’ functor $\mathcal{F}^*$ turns any group into the *underlying set* of that group. Similarly, all sorts of other ‘free’ and ‘underlying’ constructions are also left and right adjoints, respectively.

Right adjoints preserve *limits*, and left adjoints preserve *colimits*.

The category $\mathcal{C}$ is called a *cocomplete category* if every functor $\mathcal{F} : \mathcal{J} \rightarrow \mathcal{C}$ has a colimit. The following categories are cocomplete: $\mathcal{S}, \mathcal{G}, \mathcal{A}, \mathcal{T}$, and $\mathcal{PT}$.

The importance of adjoint functors lies in the fact that every functor which has a left adjoint (and therefore is a right adjoint) is continuous. In the category $\mathcal{A}$ of Abelian groups, this e.g., shows that the kernel of a product of homomorphisms is naturally identified with the product of the kernels. Also, limit functors themselves are continuous. A covariant functor $\mathcal{F} : \mathcal{J} \rightarrow \mathcal{C}$ is *cocontinuous* if it transforms colimits into colimits. Every functor which has a right adjoint (and is a left adjoint) is cocontinuous.

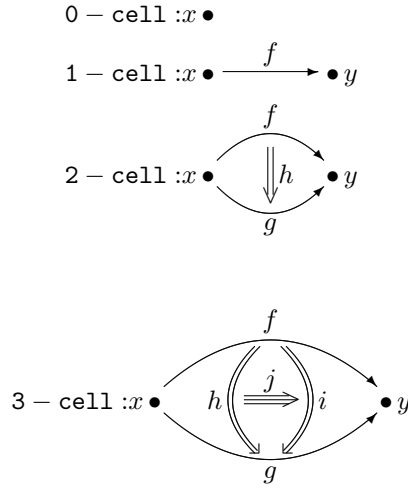
The *analogy* between *adjoint functors* and *adjoint linear operators* relies upon a deeper analogy: just as in quantum theory the inner product $\langle \phi, \psi \rangle$

represents the *amplitude* to pass from ϕ to ψ , in category theory $Mor(A, B)$ represents the *set of ways* to go from A to B . These are to Hilbert spaces as categories are to sets. The analogues of adjoint linear operators between Hilbert spaces are certain adjoint functors between 2-Hilbert spaces [Bae97, BD98]. Similarly, the *adjoint representation* of a Lie group G is the linearized version of the action of G on itself by conjugation, i.e., for each $g \in G$, the inner automorphism $x \mapsto gxg^{-1}$ gives a linear transformation $Ad(g) : \mathfrak{g} \rightarrow \mathfrak{g}$, from the Lie algebra $\mathfrak{g}$ of G to itself.

5.1.7 n -Categories and n -Functors

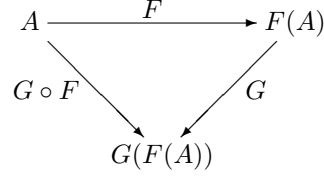
Generalization from ‘Small’ Categories to ‘Big’ n -Categories

If we think of a point in geometrical space (either natural, or abstract) as an *object* (or, a 0-cell), and a path between two points as an *arrow* (or, a 1-morphism, or a 1-cell), we could think of a ‘path of paths’ as a 2-arrow (or, a 2-morphism, or a 2-cell), and a ‘path of paths of paths’ (or, a 3-morphism, or a 3-cell), etc. Here a ‘path of paths’ is just a continuous 1-parameter family of paths from between source and target points, which we can think of as tracing out a 2D surface, etc. In this way we get a ‘skeleton’ of an n -category, where a 1-category operates with 0-cells (objects) and 1-cells (arrows, causally connecting *source* objects with *target* ones), a 2-category operates with all the cells up to 2-cells [Ben67], a 3-category operates with all the cells up to 3-cells, etc. This skeleton clearly demonstrates the *hierarchical self-similarity* of n -categories:

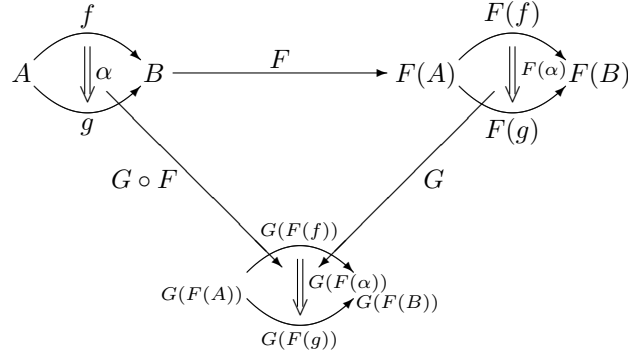


where triple arrow goes in the third direction, perpendicular to both single and double arrows. Categorical composition is defined by pasting arrows.

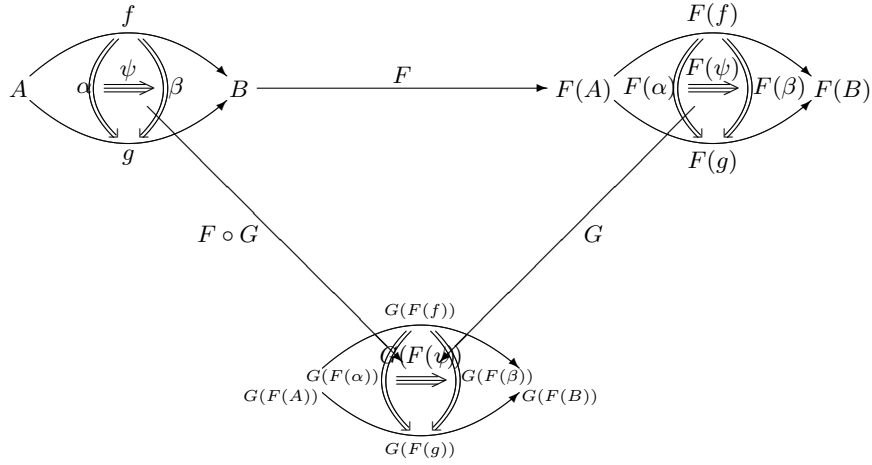
Thus, a 1-category can be depicted as a commutative triangle:



a 2–category is a commutative triangle:



a 3–category is a commutative triangle:



etc., up to n –categories.

Many deep-sounding results in mathematical sciences are get by the process of *categorification*² of the high school mathematics [CF94, BD98].

An n –category is a generic mathematical structure consisting of a collection of objects, a collection of arrows between objects, a collection of 2–arrows

² Categorification means replacing sets with categories, functions with functors, and equations between functions by natural equivalences between functors. Iterating this process requires a theory of n –categories.

between arrows [Ben67], a collection of 3-arrows between 2-arrows, and so on up to n [Bae97, BD98, Lei02, Lei03, Lei04].

More precisely, an n -category (for $n \geq 0$) consists of:

- 0-cells, or objects, $A, B, \dots$
- 1-cells, or arrows, $A \xrightarrow{f} B$, with a composition

$$A \xrightarrow{f} B \xrightarrow{g} C = A \xrightarrow{g \circ f} C$$

- 2-cells, ‘arrows between arrows’, $A \begin{array}{c} \xrightarrow{f} \\ \Downarrow \alpha \\ \xrightarrow{g} \end{array} B$, with vertical compositions

(denoted by $\circ$) and horizontal compositions (denoted by $*$), respectively given by

$$\begin{array}{c} \begin{array}{c} f \\ \Downarrow \alpha \\ A \xrightarrow{g} B \\ \Downarrow \beta \\ h \end{array} = \begin{array}{c} f \\ \Downarrow \beta \circ \alpha \\ A \xrightarrow{h} B \end{array} \quad \text{and} \\ \begin{array}{c} f \\ \Downarrow \alpha \\ A \xrightarrow{g} A' \end{array} \begin{array}{c} f' \\ \Downarrow \alpha' \\ A' \xrightarrow{g'} A'' \end{array} = \begin{array}{c} f' \circ f \\ \Downarrow \alpha' * \alpha \\ A \xrightarrow{g' \circ g} A'' \end{array} \end{array}$$

- 3-cells, ‘arrows between arrows between arrows’, $A \begin{array}{c} \xrightarrow{f} \\ \alpha \left(\begin{array}{c} \Gamma \\ \Downarrow \end{array} \right) \beta \\ \xrightarrow{g} \end{array} B$

(where the Γ -arrow goes in a direction perpendicular to f and α), with various kinds of vertical, horizontal and mixed compositions,

- etc., up to n -cells.

Calculus of n -categories has been developed as follows. First, there is $\mathcal{K}_2$, the 2-category of all ordinary (or small) categories. $\mathcal{K}_2$ has categories $\mathcal{K}, \mathcal{L}, \dots$ as objects, functors $\mathcal{F}, \mathcal{G} : \mathcal{K} \rightrightarrows \mathcal{L}$ as arrows, and natural transformations, like $\tau : \mathcal{F} \rightrightarrows \mathcal{G}$ as 2-arrows.

In a similar way, the arrows in a 3-category $\mathcal{K}_3$ are 2-functors $\mathcal{F}_2, \mathcal{G}_2, \dots$ sending objects in $\mathcal{K}_2$ to objects in $\mathcal{L}_2$, arrows to arrows, and 2-arrows to 2-arrows, strictly preserving all the structure of $\mathcal{K}_2$

$$\begin{array}{ccc}
 \begin{array}{c} f \\ \Downarrow \alpha \\ g \end{array} & \xrightarrow{\mathcal{F}_2} & \begin{array}{c} \mathcal{F}_2(f) \\ \Downarrow \mathcal{F}_2(\alpha) \\ \mathcal{F}_2(g) \end{array} \\
 A \quad B & & \mathcal{F}_2(A) \quad \mathcal{F}_2(B)
 \end{array}$$

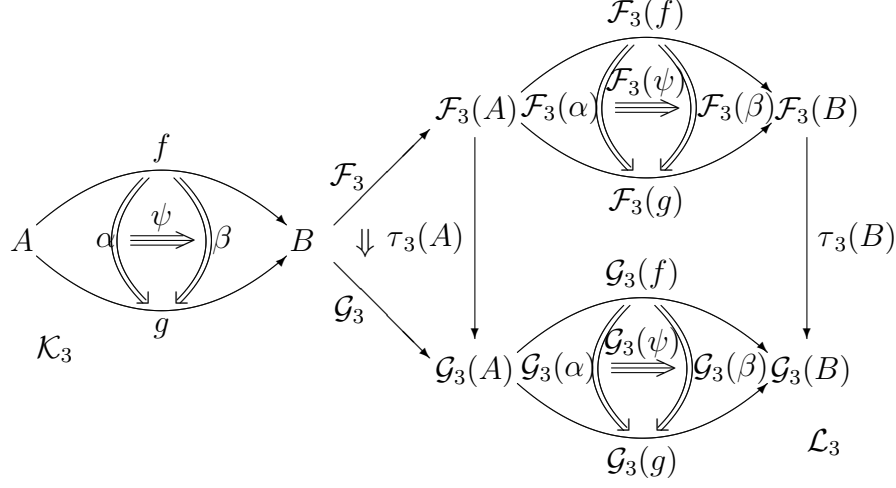
The 2-arrows in $\mathcal{K}_3$ are 2-natural transformations, like $\tau_2 : \mathcal{F}_2 \xRightarrow{2} \mathcal{G}_2$ between 2-functors $\mathcal{F}_2, \mathcal{G}_2 : \mathcal{K}_2 \rightarrow \mathcal{L}_2$ that sends each object in $\mathcal{K}_2$ to an arrow in $\mathcal{L}_2$ and each arrow in $\mathcal{K}_2$ to a 2-arrow in $\mathcal{L}_2$, and satisfies natural transformation-like conditions. We can visualize τ_2 as a prism going from one functorial picture of $\mathcal{K}_2$ in $\mathcal{L}_2$ to another, built using commutative squares:

$$\begin{array}{ccc}
 \begin{array}{c} f \\ \Downarrow \alpha \\ g \end{array} & \xrightarrow{\mathcal{F}_2} & \begin{array}{c} \mathcal{F}_2(f) \\ \Downarrow \mathcal{F}_2(\alpha) \\ \mathcal{F}_2(g) \end{array} \\
 \downarrow \tau_2(A) & & \downarrow \tau_2(B) \\
 \begin{array}{c} \mathcal{G}_2(f) \\ \Downarrow \mathcal{G}_2(\alpha) \\ \mathcal{G}_2(g) \end{array} & & \begin{array}{c} \mathcal{G}_2(A) \quad \mathcal{G}_2(B) \end{array} \\
 \mathcal{K}_2 & & \mathcal{L}_2
 \end{array}$$

Similarly, the arrows in a 4-category $\mathcal{K}_4$ are 3-functors $\mathcal{F}_3, \mathcal{G}_3, \dots$ sending objects in $\mathcal{K}_3$ to objects in $\mathcal{L}_3$, arrows to arrows, and 2-arrows to 2-arrows, strictly preserving all the structure of $\mathcal{K}_3$

$$\begin{array}{ccc}
 \begin{array}{c} f \\ \alpha \Rightarrow \beta \\ g \end{array} & \xrightarrow{\mathcal{F}_3} & \begin{array}{c} \mathcal{F}_3(f) \\ \mathcal{F}_3(\alpha) \Rightarrow \mathcal{F}_3(\beta) \\ \mathcal{F}_3(g) \end{array} \\
 A \quad B & & \mathcal{F}_3(A) \quad \mathcal{F}_3(B)
 \end{array}$$

The 2-arrows in $\mathcal{K}_4$ are 3-natural transformations, like $\tau_3 : \mathcal{F} \xRightarrow{3} \mathcal{G}$ between 3-functors $\mathcal{F}_3, \mathcal{G}_3 : \mathcal{K}_3 \rightarrow \mathcal{L}_3$ that sends each object in $\mathcal{K}_3$ to a arrow in $\mathcal{L}_3$ and each arrow in $\mathcal{K}_3$ to a 2-arrow in $\mathcal{L}_3$, and satisfies natural transformation-like conditions. We can visualize τ_3 as a prism going from one picture of $\mathcal{K}_3$ in $\mathcal{L}_3$ to another, built using commutative squares:



Topological Structure of n -Categories

We already emphasized the topological nature of ordinary category theory. This fact is even more obvious in the general case of n -categories (see [Lei02, Lei03, Lei04]).

Homotopy Theory

Any topological manifold M induces an n -category $\Pi_n(M)$ (its *fundamental n -groupoid*), in which 0-cells are *points* in M ; 1-cells are *paths* in M (i.e., parameterized continuous maps $f : [0, 1] \rightarrow M$); 2-cells are *homotopies* (denoted by $\simeq$) of *paths* relative to endpoints (i.e., parameterized continuous maps $h : [0, 1] \times [0, 1] \rightarrow M$); 3-cells are *homotopies of homotopies* of paths in M (i.e., parameterized continuous maps $j : [0, 1] \times [0, 1] \times [0, 1] \rightarrow M$); categorical *composition* is defined by *pasting* paths and homotopies. In this way the following ‘homotopy skeleton’ emerges:

$$\begin{aligned}
 0\text{-cell} &: x \bullet & x &\in M; \\
 1\text{-cell} &: x \bullet \xrightarrow{f} \bullet y & f &: x \simeq y \in M, \\
 & f : [0, 1] \rightarrow M, f : x \mapsto y, y = f(1), f(0) = x, f(1) = y; \\
 & \text{e.g., linear path: } f(t) = (1-t)x + ty;
 \end{aligned}$$

$$2\text{-cell} : x \bullet \begin{array}{c} \xrightarrow{f} \\ \Downarrow h \\ \xrightarrow{g} \end{array} \bullet y \quad h : f \simeq g \in M,$$

$h : [0, 1] \times [0, 1] \rightarrow M$, $h : f \mapsto g$, $g = h(f(x))$,
 $h(x, 0) = f(x)$, $h(x, 1) = g(x)$, $h(0, t) = x$, $h(1, t) = y$
 e.g., linear homotopy: $h(x, t) = (1 - t)f(x) + tg(x)$;

$$3\text{-cell} : x \bullet \begin{array}{c} \xrightarrow{f} \\ \left(\begin{array}{c} \Downarrow h \\ \Downarrow i \end{array} \right) j \\ \xrightarrow{g} \end{array} \bullet y \quad j : h \simeq i \in M,$$

$j : [0, 1] \times [0, 1] \times [0, 1] \rightarrow M$, $j : h \mapsto i$, $i = j(h(f(x)))$
 $j(x, t, 0) = h(f(x))$, $j(x, t, 1) = i(f(x))$,
 $j(x, 0, s) = f(x)$, $j(x, 1, s) = g(x)$,
 $j(0, t, s) = x$, $j(1, t, s) = y$
 e.g., linear composite homotopy: $j(x, t, s) = (1 - t)h(f(x)) + ti(f(x))$.

If M is a *smooth* manifold, then all included paths and homotopies need to be *smooth*. Recall that a *groupoid* is a category in which every morphism is invertible; its special case with only one object is a *group*.

Category $\mathcal{TT}$

Topological n -category $\mathcal{TT}$ has:

- 0-cells: topological spaces X
- 1-cells: continuous maps $X \xrightarrow{f} Y$

- 2-cells: homotopies h between f and $g : X \begin{array}{c} \xrightarrow{f} \\ \Downarrow h \\ \xrightarrow{g} \end{array} Y$

i.e., continuous maps $h : X \times [0, 1] \rightarrow Y$, such that $\forall x \in X$, $h(x, 0) = f(x)$ and $h(x, 1) = g(x)$

- 3-cells: homotopies between homotopies : $X \begin{array}{c} \xrightarrow{f} \\ \left(\begin{array}{c} \Downarrow h \\ \Downarrow i \end{array} \right) j \\ \xrightarrow{g} \end{array} Y$

i.e., continuous maps $j : X \times [0, 1] \times [0, 1] \rightarrow Y$.

Category $\mathcal{CK}$

Consider an n -category $\mathcal{CK}$, which has:

- 0-cells: chain complexes A (of Abelian groups, say)

- 1-cells: chain maps $A \xrightarrow{f} B$

- 2-cells: chain homotopies $A \begin{array}{c} \xrightarrow{f} \\ \Downarrow \alpha \\ \xrightarrow{g} \end{array} B$,

i.e., maps $\alpha : A \rightarrow B$ of degree 1

- 3-cells $A \begin{array}{c} \xrightarrow{f} \\ \alpha \left(\begin{array}{c} \Gamma \\ \Rightarrow \end{array} \right) \beta \\ \xrightarrow{g} \end{array} B$: homotopies between homotopies,

i.e., maps $\Gamma : A \rightarrow B$ of degree 2 such that $d\Gamma - \Gamma d = \beta - \alpha$.

There ought to be some kind of map $\mathcal{CC} : \mathcal{TT} \Rightarrow \mathcal{CK}$ (see [Lei02, Lei03, Lei04]).

Categorification

Categorification is the process of finding category-theoretic analogs of set-theoretic concepts by replacing sets with categories, functions with functors, and equations between functions by natural isomorphisms between functors, which in turn should satisfy certain equations of their own, called ‘coherence laws’. Iterating this process requires a theory of n -categories.

Categorification uses the following analogy between set theory and category theory [CF94, BD98]:

Set Theory	Category Theory
elements	objects
equations between elements	isomorphisms between objects
sets	categories
functions	functors
equations between functions	natural isomorphisms between functors

Just as sets have elements, categories have objects. Just as there are functions between sets, there are functors between categories. Now, the proper analog of an equation between elements is not an equation between objects, but an isomorphism. Similarly, the analog of an equation between functions is a natural isomorphism between functors.

5.2 Glossary of Frequent Neurophysiological Terms

This glossary is modified from [Mat01] and adapted for natural biodynamics.

A band

A darker band at the middle of a sarcomere of a striated muscle fibre, corresponding to the length of the thick filaments.

absolute refractory period

The period of time after an action potential during which a second action potential cannot be triggered. This period corresponds to the time when sodium channel inactivation gates remain closed after repolarization.

acetylcholine (ACh)

A neurotransmitter in both the central and peripheral nervous system, formed by the combination of acetate and choline, a reaction catalyzed by the synthetic enzyme choline acetyltransferase.

acetylcholinesterase

The degradative enzyme for acetylcholine, which cleaves the ester bond and releases acetate and choline.

ACh receptor

The receptor molecule that detects the presence of acetylcholine in the extracellular space and initiates the postsynaptic response to the neurotransmitter.

actin

A motor protein. Actin forms the backbone of the thin filaments of striated muscle cells and is also involved in other cellular motions, such as the movement of growth cones.

action potential (AP)

The long-distance signal of the nervous system. Action potentials are triggered in excitable cells in response to depolarization that exceeds the threshold potential.

active zone

The subregion of a presynaptic terminal where synaptic vesicles fuse with the plasma membrane, releasing their neurotransmitter content into the synaptic cleft.

ADP – Adenosine diphosphate

ADP results from hydrolysis of ATP by ATPase or kinase enzymes, with the concomitant release of the terminal phosphate group of ATP.

afferent pathway

A pathway carrying incoming sensory information into the nervous system.

afterhyperpolarization

A period after an action potential during which the membrane potential is more negative than usual.

agnosia

A collective term for the inability to identify objects based on sensory information, even though the primary sensory cortex is intact.

alpha motor neurons

Large motor neurons that innervate the extrafusal muscle fibers of skeletal muscles.

Alzheimer's disease

The most common type of age-related, progressive dementia, produced by death of neurons in a variety of brain regions. The disease is characterized by two pathological features in the brain: neuritic plaques and neurofibrillary tangles.

angular gyrus

A gyrus located at the border between the temporal, parietal, and occipital lobes of the cerebral cortex. Lesions in the angular gyrus of the language-dominant hemisphere interfere with reading and writing.

anterolateral system

The sensory projection system carrying pain and temperature information through the lateral sensory tract of the spinal cord to the brainstem, thalamus, and somatosensory cortex.

aphasia

A disorder of understanding or production of spoken language.

apraxia

A deficit in programming complex series of movements, without paralysis, sensory defects, or difficulty performing simple movements.

archicortex

A simple form of cortex having only a single layer of cells, as opposed to the multiple cell layers found in the neocortex. The hippocampus is an example of archicortex.

arcuate fasciculus

A fibre tract in the cortical white matter containing axons that interconnect Wernicke's and Broca's areas. Damage in this fibre tract produces a form of aphasia called conduction aphasia.

area MT

A portion of the middle temporal cortex (MT) involved in the detection of visual motion. Also called area V5.

area V1

Another name for the primary visual cortex, or striate cortex, located in the occipital lobe.

ATP – Adenosine triphosphate

The high-energy phosphate compound that is the primary source of energy for a variety of energy-requiring cellular processes. ATP also provides the donor phosphate group for phosphorylation of protein molecules by kinase enzymes. The second messenger, cyclic AMP, is synthesized from ATP by the enzyme adenylyl-cyclase.

autonomic nervous system

One of the major subdivisions of the nervous system, containing neurons that provide involuntary control of heart muscle, smooth muscle, and glands.

The autonomic nervous system consists of the sympathetic and parasympathetic divisions.

axon

The neurite that carries action potentials leaving the cell body of a neuron and distributes the signal to postsynaptic target cells.

basal ganglia

An interconnected group of three forebrain nuclei involved in motor control. The three nuclei are the caudate nucleus, the putamen, and the globus pallidus.

basilar membrane

A membrane within the cochlea that vibrates in response to sound stimuli. The sensory structure of the ear (the organ of Corti) rides on top of the basilar membrane.

bipolar cell

A neuron type found in the retina. Bipolar cells are second-order neurons that receive synaptic input from photoreceptors and make synapses on amacrine cells and ganglion cells.

blobs

Groups of neurons within the primary visual cortex, containing color-sensitive cells. The cells within the blobs have high numbers of mitochondria, which are revealed by staining for the mitochondrial enzyme, cytochrome oxidase.

brainstem

A collective term for the midbrain (mesencephalon) and hindbrain (rhombencephalon).

Broca's area

Region of the frontal lobe of the dominant cortical hemisphere for language, where damage produces deficits in production of speech (expressive aphasia). Broca's area is located just anterior to the portion of the primary motor cortex that controls the muscles of the mouth and tongue.

calcium-induced calcium release channel

The ion channel of the sarcoplasmic reticulum that allows calcium ions to flow out of the sarcoplasmic reticulum to trigger contraction of skeletal muscle cells.

caudate nucleus

One of the basal ganglia of the forebrain. Together with the putamen, the caudate nucleus is part of the striatum.

central sulcus

The large infolding, or groove, in the dorsal surface of the cerebral cortex, approximately midway between the anterior and posterior poles of the brain.

cerebellum

A major part of the brainstem in mammalian brains, involved in integration of sensory information and motor commands.

cerebral cortex

The outer surface of the cerebrum, containing most of the neurons in the cerebrum.

cerebral ventricles

The fluid-filled canal at the center of the brain.

cerebrospinal fluid (CSF)

The fluid filling the cerebral ventricles and the spinal canal.

cerebrum

A major subdivision of the forebrain. Together, the cerebrum and basal ganglia make up the telencephalon.

C-fibers

Small-diameter, unmyelinated axons of peripheral nerves. C-fibers carry pain and temperature information.

chemoreceptors

Sensory neurons that detect chemical sensory stimuli.

cholinergic neuron

A neuron that uses the neurotransmitter acetylcholine.

cingulate gyrus

Part of the limbic system, this outfolding of the cerebral cortex runs front to back, hidden from view within the interhemispheric fissure that separates the two cerebral hemispheres.

cingulum

An axon bundle that contains axons projecting from the cingulate gyrus to the entorhinal cortex in the limbic system.

cochlea

Part of the inner ear that contains the sensory hair cells of the sense of hearing.

cochlear nucleus

A nucleus in the brainstem that receives synaptic input from the spiral ganglion neurons of the cochlea.

collagen

A structural protein that forms the backbone of the extracellular matrix in many tissues.

cones

One of the two major classes of photoreceptors, the other being rods. Cones are less sensitive to light than rods and mediate color vision.

corpus callosum

The large fibre bundle containing axons interconnecting neurons of the two cerebral hemispheres.

corticobulbar system

The motor pathway carrying descending motor commands from the motor cortex to the brainstem.

corticospinal system

The motor pathway carrying axons from the motor cortex to the spinal cord, bypassing the brainstem motor centers.

cranial nerves

Nerves that directly connect the brain to peripheral targets. The human nervous system includes twelve distinct cranial nerves.

cross bridge

A structure visible through the electron microscope, connecting adjacent thick and thin filaments of a myofibril. The cross bridges are formed by myosin molecules that make up the thick filament.

cuneate nucleus

One of the dorsal column nuclei in the brainstem. The cuneate nucleus receives somatosensory inputs from the lateral portions of the dorsal columns of the spinal cord.

decussation

A cross over of axons from one side of the brain to the other.

dendrite

A neurite that receives synaptic inputs from other neurons.

dendritic spine

A short, hair-like projection from dendrite. Excitatory synapses commonly contact dendrites of CNS neurons at dendritic spines.

dentate nucleus

One of the deep nuclei of the cerebellum, in the brainstem.

depolarization

Movement of the membrane potential in the positive direction, from its normal negative level.

diencephalon

A subdivision of the forebrain, comprising the thalamus and hypothalamus.

dorsal column nuclei

Nuclei in the brainstem that receive synaptic inputs from somatosensory neurons whose axons are located in the dorsal columns. The dorsal column nuclei are the gracile nucleus and the cuneate nucleus.

dorsal column

A fibre bundle along the dorsal surface of the spinal cord, primarily consisting of branches of primary somatosensory neurons ascending from the spinal cord to the brainstem.

dorsal root ganglion

A ganglion located just outside the spinal cord at each vertebral segment, containing the cell bodies of sensory neurons whose axons enter the spinal cord at that segment.

dorsal root

The fibre bundle containing incoming (afferent) sensory axons entering the spinal cord at each vertebral segment.

efferent pathway

A pathway carrying outgoing motor information from neurons.

emboliform nucleus

One of the deep nuclei of the cerebellum, in the brainstem.

end-plate

The synaptic zone of a skeletal muscle cell, where postsynaptic ACh receptors are clustered at high density across from the synaptic terminal of the motor neuron.

end-plate potential

The postsynaptic electrical response elicited in a skeletal muscle cell by activation of its motor neuron.

entorhinal cortex

A part of the olfactory system and the limbic system. The entorhinal cortex provides input to the hippocampus, and receives synaptic connections from the cingulate gyrus. The entorhinal cortex is an example of paleocortex, having only two cellular layers.

epinephrine, or adrenaline

A hormone secreted by adrenal chromaffin cells into the bloodstream. Also used as a neurotransmitter.

equilibrium potential

The value of membrane potential at which a permeant ion is at equilibrium. The equilibrium potential is calculated from the *Nernst equation* (named after *Nobel Laureate Walther H. Nernst*).

excitation-contraction coupling

The process through which an action potential triggers contraction of a muscle cell.

excitatory postsynaptic potential (EPSP)

A postsynaptic change in membrane potential that promotes firing of an action potential in the postsynaptic cell, by bringing the membrane potential toward the threshold potential.

exteroceptors

Sensory receptors that detect stimuli originating outside the organism. For example, photoreceptors are exteroceptors.

extrafusal muscle fibers

The muscle cells of a skeletal muscle that generate the contractile force of the muscle and are located outside of muscle spindles.

extraocular muscles

The skeletal muscles that control the movements of the eyes.

fastigial nucleus

One of the deep nuclei of the cerebellum, in the brainstem.

forebrain

The most anterior of the three brain vesicles that arise from the neural tube early in brain development. Also, the adult brain region that arises from the forebrain vesicle. The forebrain consists of the telencephalon and diencephalon.

fornix

A large fibre bundle that contains output axons leaving the hippocampus.

fourth ventricle

The part of the cerebral ventricles located in the hindbrain.

fovea

The part of the retina that receives light from the center of the visual field, at the normal fixation point of the eye.

frontal eye field

Part of the frontal lobe where neurons generate commands that trigger saccades of the eyes.

GABA – gamma aminobutyric acid

A major inhibitory neurotransmitter in the brain. GABA is synthesized from glutamate by the enzyme glutamic acid decarboxylase.

gamma motor neuron

A special class of motor neurons that innervate the intrafusal muscle fibers within muscle spindles in skeletal muscles.

ganglion (plural: ganglia)

A cluster of neuronal cell bodies, usually located outside the central nervous system in vertebrates. Invertebrate central nervous systems consist of a series of ganglia connected by nerve bundles.

gap junction

A site of electrical connection between two cells, where gap junction channels bridge the gap between the insides of the two cells and allow small molecules such as ions to cross directly from one cell to the other.

geniculostriate pathway

The visual pathway leading from the lateral geniculate nucleus of the thalamus to the primary visual cortex (i.e., striate cortex).

glial cell

A non-neuronal cell in the nervous system that helps regulate the extracellular environment, including uptake of neurotransmitters released from synapses.

globus pallidus

One of the basal ganglia of the forebrain.

Golgi tendon organ

A specialized sensory structure that provides an indication of the force applied to a tendon by active contraction of a skeletal muscle.

gracile nucleus

One of the dorsal column nuclei in the brainstem. The gracile nucleus receives somatosensory inputs from the medial portion of the dorsal columns in the spinal cord.

gray matter

A portion of the central nervous system containing predominantly neuronal cell bodies. The relative lack of myelinated axons makes gray matter less opaque than surrounding areas containing fewer cell bodies and more myelinated fibers (white matter).

gyrus (plural: gyri)

An outfolding of the cerebral cortex in brains in which the cortex is highly folded (convoluted), such as the human brain.

habituation

Reduction in the strength of a reflexive response produced by repeated presentation of the eliciting stimulus.

hair cell

A ciliated sensory cell that produces changes in membrane potential in response to movements of the cilia induced by sensory stimuli.

hair follicle receptor

A type of sensory neuron that innervates hair follicles in the skin and is activated when the hair is deflected.

hindbrain

The most posterior of the three brain vesicles that arise at the anterior end of the neural tube during embryonic development of the brain. Also, the most posterior part of the adult brain, which develops from the hindbrain vesicle, including the pons, medulla, and cerebellum.

hippocampus

A simplified form of cerebral cortex (archicortex) located under the lower lip of the neocortex at the medial border of the temporal lobe. The hippocampus is part of the limbic system and is thought to be involved in certain types of memory.

homeostasis

The maintenance of a relatively constant internal environment despite large changes in external conditions.

horizontal cell

A neuron type found in the retina. Horizontal cells are lateral interneurons that receive synaptic inputs from photoreceptors.

hyperpolarization

A change in membrane potential in the negative direction, making the cell interior more negative.

hypothalamus

A part of the diencephalon involved in a variety of homeostatic functions, in the control of the pituitary, and in motivation and drive. The hypothalamus is a major part of the limbic system.

I band

In a striated muscle cell, the part of the sarcomere corresponding the region occupied only by thin filaments, where thin and thick filaments do not overlap.

inferior colliculus

The more posterior of the two colliculi, located on the dorsal surface of the midbrain. The inferior colliculus is a processing center for auditory information ascending from the cochlear nucleus and superior olivary nucleus in the brainstem.

inhibitory postsynaptic potential (IPSP)

A postsynaptic change in membrane potential that tends to prevent firing of an action potential in the postsynaptic cell, by bringing the membrane potential away from the threshold potential.

initial segment

The initial part of the axon as it leaves the cell body of a neuron. The initial segment is often the point at which action potentials are initiated in response to depolarization.

intermediolateral gray matter

The part of the spinal cord gray matter containing the cell bodies of sympathetic preganglionic neurons.

interneuron

A neuron in the nervous system that receives inputs from other neurons and makes synaptic contact with other neurons.

interoceptors

Sensory receptor cells that detect stimuli arising within the organism. Muscle spindle receptors are an example of interoceptors.

intrafusal muscle fibers

The specialized subset of skeletal muscle cells found within the muscle spindle.

inverse myotatic reflex

The reflex stimulated by activation of sensory neurons of Golgi tendon organs (also called the tendon organ reflex). Activation of this reflex leads to inhibition of the motor neurons for the same muscle, causing a reduction in muscle tension.

ion channel

A membrane protein that forms an aqueous pore through which charged ions can cross the membrane.

isometric contraction

A muscle contraction in which muscle length does not change although muscle tension increases.

isotonic contraction

A muscle contraction in which muscle tension remains constant during the contraction.

labyrinth

A collective term for the cochlea, semicircular canals, and otolith organs in the inner ear.

lateral column

The lateral white matter of the spinal cord.

lateral geniculate nucleus

The part of the thalamus that receives synaptic inputs from retinal ganglion cells and projects to the primary visual cortex.

lateral inhibition

Inhibition mediated by lateral interneurons, in which sensory stimuli at a particular location on the sensory surface inhibits activity in sensory pathways originating from adjacent regions of the sensory surface.

lateral intraparietal area

A part of the parietal lobe of the cerebral cortex involved in integrating visual stimuli with the control of eye movements.

lateral lemniscus

An axon tract carrying auditory information from the cochlear nucleus and superior olivary nucleus in the brainstem to the inferior colliculus in the midbrain.

lateral line organ

A sensory system found in aquatic vertebrates, containing hair cells that respond to water movement.

lateral sensory tract

Part of the lateral white matter in the spinal cord, containing axons of interneurons that receive inputs from nociceptors and temperature-sensitive sensory neurons.

lateral ventricles

The portion of the cerebral ventricles found in the telencephalon.

ligand-gated ion channel

An ion channel in which channel gating is controlled by binding of a chemical signal (the ligand) to a specific binding site on the channel protein. The ACh-gated ion channel of the neuromuscular junction is an example.

limbic system

A brain system involved in the regulation of emotion, motivation, and homeostasis.

Long-term depression (LTD)

A reduction in the strength of a synaptic connection lasting for hours or days.

Long-term potentiation (LTP)

An enhancement of postsynaptic response amplitude lasting hours or days, produced by a burst of presynaptic activity. LTP is commonly studied as a cellular model of learning and memory.

M line

In the sarcomere of a striated muscle cell, the transverse line at the midpoint of the sarcomere. The M line consists of filaments connecting the thick filaments at their midpoint.

mammillothalamic tract

A fibre tract containing axons projecting from the mammillary bodies of the hypothalamus to the thalamus in the limbic system.

marginal zone

The relatively cell free region at the outer edge of the neural tube.

mechanoreceptors

Sensory receptor neurons that respond to mechanical displacement of the sensory surface. Examples are muscle spindle receptors and hair cells.

medial geniculate nucleus

The portion of the thalamus that processes auditory information. The medial geniculate nucleus receives synaptic input from the inferior colliculus and sends axons to the primary auditory cortex.

medial lemniscus

A fibre tract carrying ascending somatosensory information from the dorsal column nuclei of the brainstem to the thalamus in the diencephalon.

medulla oblongata

The most posterior part of the brainstem, at the border between the brain and the spinal cord.

Meissner corpuscle

A rapidly adapting skin mechanoreceptor that is sensitive to touch and pressure.

membrane potential

The electrical voltage difference between the inside and the outside of a cell.

Merkel receptor

A slowly adapting skin mechanoreceptor that signals sustained pressure.

mesencephalon

The midbrain.

midbrain

The middle of the three brain vesicles that arise from the neural tube during embryonic development. In the adult brain, the midbrain consists of brain structures such as the superior colliculus, the inferior colliculus, and parts of the reticular formation.

miniature end-plate potential

A small postsynaptic depolarization at the neuromuscular junction, arising from spontaneous fusion of a single synaptic vesicle in the synaptic terminal.

motor neuron

A neuron that makes synaptic contact with the final target cell, such as a skeletal muscle cell.

motor unit

A single motor neuron and all of the muscle cells that receive synaptic connections from that motor neuron.

muscle fibre

A muscle cell.

muscle spindle

An encapsulated sensory structure activated by stretch of skeletal muscles.

myelin

The insulating sheath around axons, formed by certain types of glial cells.

myofibril

A bundle of thick and thin filaments that forms an organizational unit within a single muscle cell, which contains several myofibrils.

myosin

The protein that makes up the thick filaments of a myofibril. ATP hydrolysis by myosin provides the energy to drive filament sliding during muscle contraction.

myotatic, or stretch reflex

The spinal reflex triggered by activation of muscle spindles.

neocortex

A type of cerebral cortex characterized by multiple layers of cells. Most of the cerebral cortex in the human brain consists of neocortex. Examples

of neocortex are the primary somatosensory cortex and the primary motor cortex.

neurite

A collective term for the dendrites and axons of a neuron.

neurogenesis

The stage of neural development when neuronal precursor cells proliferate to produce neurons.

neurohypophysis

The posterior part of the pituitary gland, where nerve terminals of hypothalamic magnocellular neurosecretory cells release the hormones oxytocin and vasopressin.

neuromuscular junction

The synaptic junction between the motor neuron and its postsynaptic skeletal muscle cell.

neuron

A nerve cell.

neurotransmitter

The chemical messenger released from a synaptic terminal to influence a postsynaptic target cell.

nodes of Ranvier

Periodic breaks in the myelin sheath, where voltage-dependent sodium channels are clustered and sodium influx occurs to support action potential propagation.

norepinephrine, or noradrenaline

A neurotransmitter released by sympathetic motor neurons and by some neurons in the central nervous system.

nucleus (plural: nuclei)

A cluster of neuronal cell bodies within the central nervous system.

olfactory bulb

The part of the central nervous system that receives synaptic projections from olfactory sensory neurons, via the olfactory nerve.

oligodendrocyte

A type of glial cell that myelinates axons in the central nervous system.

optic chiasm

The cross over point of the optic nerve, where ganglion cell axons from the temporal and nasal portions of the retina are sorted to ipsilateral or contralateral projections to the lateral geniculate nucleus.

organ of Corti

The sensory structure within the cochlea, where sensory hair cells are located.

osmosis

The movement of water down its concentration gradient.

otolith organ

A sensory structure containing hair cells that detect the organism's orientation with respect to gravity.

Pacinian corpuscle

A rapidly adapting skin mechanoreceptor that is sensitive to touch and pressure.

paleocortex

A form of cerebral cortex characterized by two layers of cells, as opposed to the multiple layers of cells found in the neocortex. Entorhinal cortex of the olfactory system is an example of paleocortex.

Papez circuit

The central core of the limbic system, consisting of a loop from the cingulate gyrus, entorhinal cortex, hippocampus, hypothalamus, thalamus, and back to the cingulate gyrus.

parasympathetic division

The acetylcholine-releasing division of the autonomic nervous system.

paravertebral ganglia

The chain of sympathetic ganglia that parallel the spinal column.

Parkinson's disease

A human disease characterized by muscle tremor and difficulty in initiating and sustaining locomotion. The disease results from degeneration of dopamine-releasing neurons of the substantia nigra.

pituitary gland

A master control endocrine gland at the base of the brain. The pituitary gland is controlled in turn by the hypothalamus.

pons

A major subdivision of the hindbrain.

postcentral gyrus

The gyrus located just posterior to the central sulcus, consisting of the primary somatosensory cortex.

postsynaptic cell

The target cell at a synapse.

post-tetanic potentiation

Synaptic potentiation that follows a sustained, high-frequency burst of presynaptic action potentials.

precentral gyrus

The gyrus located just anterior to the central sulcus, consisting of the primary motor cortex.

premotor cortex

Part of the frontal lobe anterior to the primary motor cortex, containing neurons that encode complex movements.

preoptic area

Part of the telencephalon just anterior and superior to the anterior end of the hypothalamus. The preoptic area is closely associated with the hypothalamus and is usually considered to be part of the hypothalamus.

presynaptic cell

The input cell at a synapse.

prevertebral ganglia

Sympathetic ganglia located in the abdominal cavity.

primary visual cortex

The visual cortical area that receives direct input from the lateral geniculate nucleus. The primary visual cortex (area V1; striate cortex) is located at the posterior pole of the occipital lobe.

process

Another name for a neurite.

proprioceptor

A sensory receptor neuron that detects limb or joint position, muscle length, or muscle tension.

prosencephalon

The forebrain.

Purkinje cell

The output cells of the cerebellum.

putamen

One of the basal ganglia of the forebrain. Together, the putamen and the caudate nucleus form the striatum.

pyramidal cell

A type of cortical neuron shaped like a pyramid, with a long apical dendrite originating from the narrow end of the cell.

pyramids

The fibre bundles consisting of descending axons from the primary motor cortex. radial glial cell. A glial cell that extends from the ventricular zone to the marginal zone during early neural development. Migrating neurons leaving the ventricular zone follow the long thin radial glial cells.

raphe nucleus

A nucleus located near the midline of the brainstem, containing (among other neurons) the omnidirectional pause neurons that allow saccades to proceed.

receptive field

The portion of the sensory surface where stimuli affect the activity of a sensory neuron.

receptor potential

The change in membrane potential in a primary sensory receptor neuron in response to a sensory stimulus.

red nucleus

A brainstem motor control nucleus that gives rise to the rubrospinal tract. Activation of the rubrospinal tract promotes limb flexion.

Renshaw cell

An inhibitory interneuron in the spinal cord that receives excitatory input from a motor neuron and makes inhibitory synapses back onto the same motor neuron.

resting potential

The steady state membrane potential of a neuron in the absence of incoming synaptic or sensory influences.

reticular formation

A diffuse network of neurons in the midbrain and hindbrain, involved in a variety of sensory and motor functions.

reticulospinal tract

A fibre tract consisting of descending axons from neurons in the reticular formation to spinal interneurons and motor neurons.

retina

The multilayered structure at the back of the eye responsible for light reception and processing of visual information. The retina consists of the neural retina, containing the neurons and glial cells, and the retinal pigmented epithelium, which absorbs stray light and supports the outer segments of photoreceptor cells.

retinal (retinaldehyde)

The light-absorbing chromophore group that is chemically attached to the opsin protein to form a visual pigment molecule.

retinal ganglion cell

The output cells of the retina, whose axons form the optic nerve and project to the lateral geniculate nucleus of the thalamus, the accessory optic system, and the suprachiasmatic nucleus of the hypothalamus.

retinohypothalamic tract

The fibre tract consisting of axons of retinal ganglion cells projecting to the suprachiasmatic nucleus of the hypothalamus.

rhodopsin

The visual pigment molecule of rod photoreceptors.

rhombencephalon

The hindbrain.

rods

A subtype of photoreceptor found in the vertebrate retina. Rods are more sensitive to light than cones are responsible for vision under dim illumination.

rubrospinal tract

The fibre tract containing axons descending to the spinal cord from the red nucleus of the brainstem.

Ruffini corpuscle

A slowly adapting skin mechanoreceptor that signals sustained pressure.

saccade

A rapid eye movement used to alter eye position within the orbit, causing a rapid adjustment of the fixation point to different positions in the visual world.

saccule

The horizontally oriented otolith organ of the labyrinth.

saltatory conduction

A form of action potential propagation found in myelinated axons, in which action potentials jump from one node of Ranvier to the next.

sarcomere

The basic repeating unit of striation along the length of myofibrils of striated muscle cells. The sarcomere is defined as extending from one Z line to the next Z line.

sarcoplasmic reticulum

An intracellular store of calcium ions wrapped around the contractile apparatus of myofibrils in striated muscle cells. Calcium released from the sarcoplasmic reticulum triggers contraction.

Schwann cell

A type of glial cell that forms the myelin sheath around axons in the peripheral nervous system.

semicircular canals

Bio-accelerometers (i.e., acceleration-sensing organs), which are fluid-filled loops that form part of the labyrinth.

sensitization

Enhancement of the strength of a reflexive response produced by the presentation of a noxious stimulus.

sensory adaptation

The reduction in activity of a sensory neuron during sustained application of a sensory stimulus.

sensory neuron

A neuron whose activity is affected by the presence of a particular type of sensory stimulus.

sensory transduction

The conversion of stimulus energy into an electrical signal in a primary sensory receptor neuron.

skeletal muscle

A type of striated muscle responsible for movement of body parts.

soma

The cell body of a cell, where the nucleus is located.

somatic nervous system

The division of the nervous system that controls the skeletal muscles, as distinguished from the autonomic nervous system.

somatopic map

A form of neural organization in which neighboring regions of a body structure project to or are controlled by neighboring neurons in the brain region that receive sensory inputs from or send motor output to the body structure.

somatosensory system

The sensory system that receives and processes sensory information from the body, including the skin, muscles, and joints.

spatial summation

Summation of postsynaptic responses in a postsynaptic cell from two or more synaptic inputs that are active at about the same time.

spinal canal

The fluid filled space at the center of the spinal cord. The spinal canal is continuous with the ventricles of the brain.

spinal nerve

The mixed motor and sensory nerve connected to the spinal cord at a particular vertebral segment.

spinocerebellar tract

The sensory pathway in the spinal cord carrying ascending axons to the cerebellum.

spiral ganglion

The ganglion in the cochlea containing the cell bodies of sensory neurons that receive inputs from the cochlear hair cells and send axons via the auditory nerve (cranial nerve VIII) to the cochlear nucleus of the brainstem.

stellate cell

A neuron whose dendrites radiate approximately equally in all directions from the soma, producing a starlike pattern.

stem cell

An undifferentiated precursor cell that retains the ability to give rise to a variety of cell types. In the nervous system, stem cells can give rise to various neuron subtypes and to glial cells.

striated muscle cell

A type of muscle cell in which the contractile machinery forms a regular, repeating array, which gives the cell a striped (striated) appearance when viewed through the light microscope.

striatum

A collective term for the caudate nucleus and putamen, which are two of the basal ganglia of the forebrain.

substantia nigra

A midbrain region involved in the control of motor behavior. Loss of dopamine-releasing neurons of the substantia nigra underlies the movement disorder called Parkinson's disease.

sulcus (plural: sulci)

An infolding, or groove, in the cortical surface. A sulcus separates neighboring gyri.

superior colliculus

A brain region on the dorsal surface of the midbrain that is involved in the control of eye movements.

superior olivary nucleus (superior olive)

A nucleus in the brainstem that is involved in the processing of auditory information. The superior olivary nucleus receives inputs from the cochlear nuclei and sends outputs to the inferior colliculus.

supplemental motor area

A higher order cortical motor area located in the medial part of the frontal lobe, just anterior to the primary motor cortex.

suprachiasmatic nucleus

A nucleus of the hypothalamus responsible for synchronizing circadian rhythms in other organs and tissues.

Sylvian fissure

The large sulcus that separates the temporal lobe from the rest of the cerebral cortex. It is also called the lateral sulcus.

sympathetic chains

A series of interconnected sympathetic nuclei (the paravertebral ganglia) that parallel both sides of the vertebral column.

sympathetic division

A division of the autonomic nervous system, containing autonomic motor neurons that release the neurotransmitter norepinephrine. Actions of the sympathetic nervous system typically oppose the actions of the other division of the autonomic nervous system, the parasympathetic division.

synapse

The contact point where a neuron transfers information to a target cell.

synaptic cleft

The extracellular space separating the presynaptic cell and the postsynaptic cell at a synapse.

synaptic vesicle

A small, membrane-bound structure in which neurotransmitter molecules are stored within synaptic terminals.

taste bud

A cluster of cells on the surface of the tongue, containing taste receptor cells.

tectorial membrane

A sheet of tissue overlying the organ of Corti in the cochlea. Cilia of outer hair cells are embedded into the surface of the tectorial membrane.

telencephalon

A subdivision of the forebrain, comprising the cerebrum and the basal ganglia.

temporal summation

Summation of successive postsynaptic responses in a postsynaptic cell from two or more action potentials arriving within a brief period in the same synaptic terminal.

tetrodotoxin

A biological toxin that blocks voltage-dependent sodium channels.

thalamus

One of the two subdivisions of the diencephalon. The thalamus receives and processes sensory information and sends the sensory information to the appropriate regions of the cerebral cortex. The thalamus also plays important roles in motor control.

thermoreceptors

Primary sensory neurons that respond to changes in skin or body temperature. thick filament. A longitudinal filament found in striated muscle cells, made up of the protein myosin.

thin filament

A longitudinal filament found in striated muscle cells, made up of the protein actin and the associated proteins tropomyosin and troponin.

third ventricle

The part of the brain ventricles that extends from the midbrain through the diencephalon.

threshold (threshold potential)

The value of membrane potential that must be reached in an excitable cell to trigger an action potential.

tonotopic map

The orderly projection of inputs originating from the cochlea to sensory areas in the brain, such that neighboring neurons in the target regions respond to progressively higher frequencies.

transverse tubules

Invaginations of the plasma membrane in skeletal muscle cells that provide a path for depolarization during the muscle action potential to spread to the cell interior.

tropomyosin

A protein associated with the thin filaments of striated muscle cells. Tropomyosin controls the access of myosin to the myosin binding site of actin.

troponin

A calcium binding molecule associated with the thin filaments of striated muscle cells. Binding of calcium to troponin initiates contraction.

tubulin

A protein molecule that polymerizes to form the backbone of microtubules. tympanic membrane. The eardrum, which transfers sound pressure waves to the bones of the middle ear.

undershoot

The transient period of increased negativity at the termination of an action potential.

utricle

The vertically oriented otolith organ of the labyrinth.

ventral column

The white matter on the ventral surface of the spinal cord, containing descending motor axons of the corticospinal tract, the vestibulospinal tract, and the reticulospinal tract.

ventral corticospinal tract

The portion of the ventral column containing descending axons of neurons whose cell bodies are located in the primary motor cortex.

ventral root

The fibre bundle containing outgoing (efferent) motor axons exiting the spinal cord at each vertebral segment.

ventricles

The fluid-filled core of the brain, filled with cerebrospinal fluid.

ventricular zone

The inner layer of the neural tube, next to the fluid-filled ventricle. Dividing precursor cells that give rise to the nervous system are found in the ventricular zone.

vestibular ganglion

A ganglion located just outside the labyrinth of the inner ear, containing the cell bodies of sensory neurons that receive inputs from the hair cells of the semicircular canals.

vestibular nuclei

Nuclei in the brainstem that receive synaptic inputs from sensory neurons of the vestibular ganglion.

vestibulo-ocular reflex

The reflex that induces eye movements in response to head rotation to keep the eyes fixated at a constant point in space.

vestibulospinal tract

A fibre pathway originating in the vestibular nucleus of the brainstem and projecting to the spinal cord. Activation of the vestibulospinal tract promotes limb extension.

voltage-sensitive sodium channel

A sodium channel whose conducting state depends on voltage. Opening of voltage-sensitive sodium channels underlies the depolarizing phase of the action potential.

Wernicke's area

Part of the temporal lobe surrounding the primary auditory cortex. Damage in Wernicke's area produces deficits in understanding spoken language (receptive aphasia).

white matter

Regions of the central nervous system containing few neuronal cell bodies and many myelinated axons. The myelin sheaths are opaque compared to surrounding regions containing mostly neuronal cell bodies (gray matter).

Z line

A crosswise line connecting thin filaments within a myofibril. The sarcomere is defined as extending from one Z line to the next Z line.

References

- AA68. Arnold, V.I., Avez, A.: Ergodic Problems of Classical Mechanics. Benjamin, New York, (1968)
- AA80. Aubry, S., André, G.: Colloquium on Computational Methods in Theoretical Physics. In Group Theoretical Methods in Physics, Horowitz and Ne'eman (ed), Ann. Israel Phys. Soc. **3**, 133–164, (1980)
- AAC90. Artuso, R., Aurell, E., Cvitanovic, P.: Recycling of strange sets: I & II. Nonlinearity **3**, 325–359 and 361, (1990)
- AAM76. Anderson, B.D., Arbib, M.A., Manes, E.G.: Foundations of System Theory: Finitary and Infinitary Conditions. Lecture Notes in Economics and Mathematical Systems Theory, Springer, New York, (1976)
- ABC96. Aurell, E., Boffetta, G., Crisanti, A., Paladin, G., Vulpiani, A.: Predictability in Systems with Many Characteristic Times: The Case of Turulence. Phys. Rev. E **53**, 2337, (1996)
- ABC97. Artale, V., Boffetta, G., Celani, A., Cencini, M., Vulpiani, A.: Dispersion of passive tracers in closed basins: Beyond the diffusion coefficient. Phys. Fluids A **9**, 3162, (1997)
- ABM93. Abeles, M., Bergman, H., Margalit, E., Vaadia, E.: Spatiotemporal firing patterns in the frontal cortex of behaving monkeys. J. Neurophysiol. **70**, 1629–1638, (1993)
- ABS00. J. A. Acebrón, L. L. Bonilla, and R. Spigler: Synchronization in populations of globally coupled oscillators with inertial effects. Phys. Rev. E **62**, 34373454, (2000)
- AC91. Ablowitz, M.J., Clarkson, P.A.: Solitons, nonlinear evolution equations and inverse scattering. London Math. Soc., **149**, CUP, Cambridge, UK, (1991)
- ACF98. Arena, P., Caponetto, R., Fortuna, L., Porto, D.: Bifurcation and chaos in non-integer order cellular neural networks. Int. J. Bif. Chaos **8**, 1527–1539, (1998)
- AEH05. Ahmed, E., Elgazzar, A.S., Hegazi, A.S.: An Overview of Complex Adaptive Systems. Mansoura J. Math. (to appear)

- AF84. Alfsen, K.H., Frøyland, J.: Systematics of the Lorenz model at $\sigma = 10$. Technical report, University of Oslo, Inst. of Physics, Norway, (1984)
- AFH86. Albeverio, S., Fenstat, J., Hoegh-Krohn, R., Lindstrom, T.: Nonstandard Methods in Stochastic Analysis and Mathematical Physics. Academic Press, New York, (1986)
- AG05. Apps, R., Garwicz, M.: Anatomical and physiological foundations of cerebellar information processing. *Nature Rev. Neurosci.*, **6**, 297–311, (2005)
- AGI98. Amaral, L.A.N., Goldberger, A.L., Ivanov, P.Ch., Stanley, H.E.: Scale-independent measures and pathologic cardiac dynamics. *Phys. Rev. Lett.*, **81**, 2388–2391, (1998)
- AGM94. Alekseevsky, D.V., Grabowski, J., Marmo, G., Michor, P.W.: Poisson structures on the cotangent bundle of a Lie group or a principle bundle and their reductions. *J. Math. Phys.*, **35**, 4909–4928, (1994)
- AGM97. Alekseevsky, D., Grabowski, J., Marmo, G., Michor, P.W.: Completely integrable systems: a generalization. *Mod. Phys. Let. A*, **12**(22), 1637–1648, (1997)
- AGS85a. Amit, D.J., Gutfreund, H., Sompolinsky, H.: Spin-glass models of neural networks. *Phys. Rev. A* **32**, 10071018, (1985)
- AGS85b. Amit, D.J., Gutfreund, H., Sompolinsky, H.: Storing Infinite Numbers of Patterns in a Spin-glass Model of Neural Networks. *Physical Review Letters*, **55**, 1530–1533, (1985)
- AGS87. Amit, D., Gutfreund, H., Sompolinsky, H.: Information storage in neural networks with low levels of activity. *Phys. Rev. A* **35**, 2293–2303, (1987)
- AGS95. Amit, D., Gutfreund, H., Sompolinsky, H.: Storing Infinite Numbers of Patterns in a Spin-Glass Model of Neural Networks. *Phys. Rev. Lett.* **55**, 1530–1533, (1985)
- AH75. Atherton, R.W., Homsy, G.M.: On the existence and formulation of variational principles for nonlinear differential equations, *Stud. Appl. Math.*, **54**, 31–60, (1975)
- AJB00. Albert R., Jeong H. and Barabási A.-L.: Attack and error tolerance of complex networks. *Nature* **406**, 378–382, (2000)
- AJB99. Albert, R., Jeong, H., Barabasi, A.-L.: Diameter of the world wide web. *Nature*, **401**, 130–131, (1999)
- AKO03. Ansumali, S., Karlin, I.V., Öttinger, H.C.: Minimal Entropic Kinetic Models for Hydrodynamics, *Europhys. Lett.* **63**, 798–804, (2003)
- AKO99. Aonishi, T., Kurata, K., Okada, M.: Statistical Mechanics of an Oscillator Associative Memory with Scattered Natural Frequencies. *Phys. Rev. Lett.* **82**, 2800–2803, (1999)
- AKS04. Ansumali, S., Karlin, I.V., Succi, S.: Kinetic theory of turbulence modeling: smallness parameter, scaling and microscopic derivation of Smagorinsky model. *Physica A* **338**, 379, (2004)
- AL05. Achimescu, S., Lipan, O.: Signal Propagation in Nonlinear Stochastic Gene Regulatory Networks. 3rd Int. Conf. Path. Netw., Sys. Rhodes, Greece 2005.

- AL91. Aidman, E.V., Leontiev, D.A.: From being motivated to motivating oneself: a Vygotskian perspective. *Stud. Sov. Thought*, **42**, 137–151, (1991)
- AM88. Amari, S.I., Maginu, K.: Statistical neurodynamics of associative memory. *Neu. Net.* **1**, 63–73, (1988)
- AM91. Aringazin, A., Mikhailov, A.: Matter fields in space–time with vector non–metricity. *Clas. Quant. Grav.* **8**, 1685, (1991)
- AMR88. Abraham, R., Marsden, J., Ratiu, T.: *Manifolds, Tensor Analysis and Applications*. Springer, New York, (1988)
- AN00. Amari, Nagaoka, H.: *Methods of Information Geometry*, Oxford Univ. Press and Amer. Math. Soc., (2000)
- AN99. Aoyagi, T., Nomura, M.: Oscillator Neural Network Retrieving Sparsely Coded Phase Patterns. *Phys. Rev. Lett.* **83**, 1062–1065, (1999)
- ANW03. Amari, S., Nakahara, H., Wu, S., Sakai, Y.: Synfiring and higher-order interactions in neuron pool. *Neural Comp.* **15**, 127–142, (2003)
- AR95. Antoni, M., Ruffo, S.: Clustering and relaxation in long-range Hamiltonian dynamics. *Phys. Rev. E*, **52**, 2361–2374, (1995)
- AS72. Abramowitz, M., Stegun, I.A.: *Handbook of Mathematical Functions*. Dover, New York, (1972)
- AS92. Abraham, R., Shaw, C.: *Dynamics: the Geometry of Behavior*. Addison–Wesley, Reading, MA, (1992)
- ATT90. Aihara, K., Takabe, T., Toyoda, M.: Chaotic Neural Networks, *Phys. Lett. A* **144**, 333–340, (1990)
- AV00. Alfinito, E., Vitiello, G.: Formation and life–time of memory domains in the dissipative quantum model of brain, *Int. J. Mod. Phys. B*, **14**, 853–868, (2000)
- Aba96. Abarbanel, H.D.I.: *Analysis of Observed Chaotic Data*. Springer-Verlag, Berlin, (1996)
- Abe91. Abeles, M.: *Corticonics*. Cambridge Univ. Press, Cambridge, (1991)
- Ach97. Acheson, D.: *From Calculus to Chaos*. Oxford Univ. Press, Oxford, (1997)
- Ada78. Adams, J.F.: *Infinite Loop Spaces*. Princeton Univ. Press, Princeton, NJ, (1978)
- Ahm93. Ahmad, S.: On the nonautonomous Volterra–Lotka competition equations. *Proc. Amer. Math. Soc.*, **117**, 199–205, (1993)
- Ala99. Alaoui, A.M.: Differential Equations with Multispiral Attractors. *Int. Journ. of Bifurcation and Chaos*, **9**(6), 1009–1039, (1999)
- Ale68. Alekseev, V.: Quasi-random dynamical systems I, II, III. *Math. USSR Sbor.* **5**, 73–128, (1968)
- Alo03. Alon, U.: Biological Networks: The tinkerer as an engineer. *Science* **301**, 1866–1867, (2003)
- Alt95. Von Altrock, C.: *Fuzzy Logic and Neuro-Fuzzy Applications Explained*. Prentice Hall, Englewood Cliffs, NJ, (1995)
- Ama77. Amari, S.I.: Neural theory of association and concept-formation. *Biol Cybern.* **26**(3), 175–185, (1977)

- Ama85. Amari, S.I.: Differential Geometrical Methods in Statistics. Springer, New York, (1985)
- Ama88. Amari, S.: Statistical neurodynamics of various versions of correlation associative memory Proceedings of IEEE Conference on Neural Networks **1**, 633, (1988)
- Ama89. Amari, S.: Characteristics of sparsely encoded associative memory. Neu. Net. **2**, 451-457, (1989)
- Ame93. Amemiya, Y.: On nonlinear factor analysis. Proc. Social Stat. Section. Ann. Meet. Ame. Stat. Assoc. 290-294, (1993)
- Ami89. Amit, D.J.: Modeling brain function: the world of attractor neural networks. Cambridge Univ. Press, Cambridge, (1989)
- Ano80. Anochin, P.K.: Theory of Functional Systems, (in Russian). Science, Moscow, (1980)
- Arb66. Arbib, M.A.: Categories of (M, R) -systems. Bull. Math. Biol., **28**, 511-517, (1966)
- Arb87. Arbib, M.: Brains, Machines and Mathematics. Springer, (1987)
- Arb95. Arbib, M.A. (ed.): The Handbook of Brain Theory and Neural Networks. MIT Press, (1995)
- Arb98. Arbib, M. (ed.): Handbook of Brain Theory and Neural Networks (2nd ed.). MIT Press, Cambridge, MA, (1998)
- Ark01. Arkin, A.P.: Synthetic cell biology. Curr. Opin. Biotech., **12**, 638-644, (2001)
- Arn61. Arnold, V.I.: On the stability of positions of equilibrium of a Hamiltonian system of ordinary differential equations in the general elliptic case. Sov. Math. Dokl. **2**, 247, (1961)
- Arn65. Arnold, V.I.: Sur une propriété topologique des applications globalement canoniques de la mécanique classique. C. R. Acad. Sci. Paris A, **261**, 17, (1965)
- Arn78. Arnold, V.I.: Ordinary Differential Equations. MIT Press, Cambridge, MA, (1978)
- Arn88. Arnold, V.I.: Geometrical Methods in the Theory of Ordinary differential equations. Springer, New York, (1988)
- Arn89. Arnold, V.I.: Mathematical Methods of Classical Mechanics (2nd ed). Springer, New York, (1989)
- Arn92. Arnold, V.I.: Catastrophe Theory. Springer, Berlin, (1992)
- Arn93. Arnold, V.I.: Dynamical systems. Encyclopaedia of Mathematical Sciences, Springer, Berlin, (1993)
- Ash94. Ashcraft, M.H.: Human Memory and Cognition (2nd ed). HarperCollins, New York, (1994)
- Asi04. Asif, A.: Fast implementations of the Kalman-Bucy filter for satellite data assimilation. IEEE Sig. Proc. Let., **11**(2), 235-238, (2004)
- Ati88. Atiyah, M.F.: Topological quantum field theory. Publ. Math. IHES **68**, 175-186, (1988)
- Att71. Attneave, F.: Multistability in Perception. Scientific American, **225**, 62-71, (1971)
- Ave99. Averin, D.V.: Solid-state qubits under control. Nature **398**, 748-749, (1999)

- Axe76. Axelrod, R.: *Structure of Decision, The Cognitive Maps of Political Elites*. Princeton Univ. Press, Princeton, NJ, (1976)
- B-Y97. Bar-Yam, Y.: *Dynamics of Complex Systems*. Perseus Books, Reading, (1997)
- BA99. Barabási, A-L., Albert, R.: Emergence of Scaling in Random Networks. *Science* **286**, 509, (1999)
- BBR91. Birmingham, D., Blau, M., Rakowski, M., Thompson, G.: Topological field theory. *Phys. Rep.* **209**, 129, (1991)
- BBV04. Bolle, D., Blanco, J.B., Verbeiren, T.: The signal-to-noise analysis of the Little-Hopfield model revisited. *J. Phys. A: Math. Gen.* **37**(6), 1951-1969, (2004)
- BBV99. Béguin, F., Bonatti, C., Vieitez, J.L.: Construction de flots de Smale en dimension 3. *Ann. Fac. Sci. de Toulouse Math.* **6**, 369-410, (1999)
- BC91. Benedicks, M., Carleson, L.: Dynamics of the Hénon map. *Ann. of Math.* **133**, 73-169, (1991)
- BCB92. Borelli, R.L., Coleman, C., Boyce, W.E.: *Differential Equations Laboratory Workbook*. Wiley, New York, (1992)
- BCF02. Boffetta, G., Cencini, M., Falcioni, M., Vulpiani, A.: Predictability: a way to characterize Complexity, *Phys. Rep.*, **356**, 367-474, (2002)
- BCI93. Balmfort, N.J., Cvitanovic, P., Ierley, G.R., Spiegel, E.A., Vattay, G.: Advection of vector-fields by chaotic flows. *Ann. New York Acad. Sci.* **706**, 148, (1993)
- BCR82. Borsellino, A., Carlini, F., Riani, M., Tuccio, M.T., Marco, A.D., Penengo, P., Trabucco, A.: Effects of visual angle on perspective reversal for ambiguous patterns. *Perception.* **11**, 263-273, (1982)
- BD02. Busemeyer, J.R., Diederich, A.: Survey of decision field theory. *Math. Soc. Sci.*, **43**, 345-370, (2002)
- BD95. Baez, J., Dolan, J.: Higher dimensional algebra and topological quantum field theory. *J. Math. Phys.* **36**, 6073-6105, (1995)
- BD98. Baez, J., Dolan, J.: Higher-Dimensional Algebra III: n -categories and the Algebra of Opetopes. *Adv. Math.* **135**(2), 145-206, (1998)
- BDA00. Bollé, D., Dominguez, D.R.C., Amari, S.: Mutual Information of Sparsely Coded Associative Memory with Self-Control and Ternary Neurons. *Neural Networks* **13**, 455-462, (2000)
- BDG93. Bielawski, S., Derozier, D., Glorieux, P.: Experimental characterization of unstable periodic orbits by controlling chaos. *Phys. Rev. A*, **47**, 2492, (1993)
- BDJ89. Barhen, J., Dress, W.B., and Jorgensen, C.C.: Applications of Concurrent Neuromorphic Algorithms for Autonomos Robots. In *Neural computers* (ed. Eckmiller, R., Malsburg, C.) Springer, Berlin, (1989)
- BDR01. Barré, J., Dauxois, T., Ruffo, S.: Clustering in a Hamiltonian with repulsive long range interactions. *Physica A* **295**, 254 (2001)

- BDS89. Buhmann, J., Divko, R., Schulten, K.: Associative memory with high information content. *Phys. Rev. A* **39**, 2689-2692, (1989)
- BF71. Bransford, J.D., Franks, J.J.: The Abstraction of Linguistic Ideas. *Cogn. Psych.*, **2**, 331-350, (1971)
- BFL02. Barbara, P., Filatrella, G., Lobb, C.J., Pederson, N.F.: *Studies of High Temperature Superconductors*, NOVA Science Publishers **40**, Huntington, (2002)
- BFR04. Bagnoli, F., Franci, F., Rechtman, R.: Chaos in a simple cellular automata model of a uniform society. In *Lec. Not. Comp. Sci.*, Vol. 3305, 513-522, Springer, London, (2004)
- BG90. Bouchaud, J.P., Georges, A.: Anomalous diffusion in disordered media - statistical mechanisms, models and physical applications. *Phys. Rep.* **195**, 127, (1990)
- BG96. Baker, G.L., Gollub, J.P.: *Chaotic Dynamics: An Introduction* (2nd ed). Cambridge Univ. Press, Cambridge, (1996)
- BGG80. Benettin, G., Giorgilli, A., Galgani, L., Strelcyn, J.M.: Lyapunov exponents for smooth dynamical systems and for Hamiltonian systems; a method for computing all of them. Part 1: theory, and Part 2: numerical applications. *Meccanica*, **15**, 9-20 and 21-30, (1980)
- BGL00. Boccaletti, S., Grebogi, C., Lai, Y.-C., Mancini, H., Maza, D.: *The Control of Chaos: Theory and Applications*. *Physics Reports* **329**, 103-197, (2000)
- BGS76. Benettin, G., Galgani, L., Strelcyn, J.M.: Kolmogorov Entropy and Numerical Experiments. *Phys. Rev. A* **14** 2338, (1976)
- BH96. Banaszuk, A., Hauser, J.: Approximate feedback linearization: a homotopy operator approach. *SIAM J. Cont. & Optim.*, **34**(5), 1533-1554, (1996)
- BJH95. Le Bihan, D., Jezard, P., Haxby, J., Sadato, N., Rueckert, L., Mattay, V.: Functional magnetic resonance imaging of the brain. *Ann. Intern. Med.*, **122**(4), 296-303, (1995)
- BK64. Bellman, R., Kalaba, R.: *Selected papers on mathematical trends in control theory*. Dover, New York, (1964)
- BK92. Berleant, D., Kuipers, B.: Qualitative-Numeric Simulation with Q3, in *Recent Advances in Qualitative Physics*. eds. Boi Faltings and Peter Struss, MIT Press, Cambridge, (1992)
- BKO98. De Berg, M., van Kreveld, M., Vermars, M.O., Schwarzkopf, O.: *Computational Geometry, Algorithms and Applications*. Springer Verlag, (1998)
- BL92. Blackmore, D.L., Leu, M.C.: Analysis of swept volumes via Lie group and differential equations. *Int. J. Rob. Res.*, **11**(6), 516-537, (1992)
- BLV01. Boffetta, G., Lacorata, G., Vulpiani, A.: Introduction to chaos and diffusion. *Chaos in geophysical flows*, ISSAOS, (2001)
- BM00. Choquet-Bruhat, Y., DeWitt-Morete, C.: *Analysis, Manifolds and Physics, Part II: 92 Applications* (rev. ed). North-Holland, Amsterdam, (2000)

- BM82. Choquet-Bruhat, Y., DeWitt-Morete, C.: *Analysis, Manifolds and Physics* (2nd ed). North-Holland, Amsterdam, (1982)
- BMA72. Borsellino, A., Marco, A. D., Allazatta, A., Rinsei, S. Bartolini, B.: Reversal time distribution in the perception of visual ambiguous stimuli. *Kybernetik*, **10**, 139–144, (1972)
- BMB82a. Van den Broeck, C., Mansour, M., Baras, F.: Asymptotic Properties of Coupled Non-linear Langevin Equations in the Limit of Weak Noise, I: Cusp Bifurcation. *J. Stat. Phys.* **28**, 557–575, (1982)
- BMB82b. Van den Broeck, C., Mansour, M., Baras, F.: Asymptotic Properties of Coupled Non-linear Langevin Equations in the Limit of Weak Noise, II: Transition to a Limit Cycle. *J. Stat. Phys.* **28**, 577–587, (1982)
- BMB94. Bassar, P.J., Mattiello, J., Le Bihan, D.: MR diffusion tensor spectroscopy and imaging. *Biophys. J.* **66**, 259–267, (1994)
- BMP01. LeBihan, D., Mangin, J.F., Poupon, C., Clark, C.A., Pappata, S., Molko, N., Chabriet, H.: Diffusion tensor imaging: concepts and applications. *J. Magn. Reson. Imaging*, **13**(4), 534–46, (2001)
- BMP97. Breckenridge, J.C., Myers, R.C., Peet, A.W., Vafa, C.: D-branes and spinning black holes. *Phys. Lett. B* **391**, 93, (1997)
- BMT04. Basu, S., Mehreja, R., Thiberge, S., Chen, M.T., Weiss, R.: Spatiotemporal control of gene expression with pulse generating networks. *Proc Natl. Acad. Sci. USA*, **101**, 6355–6360, (2004)
- BO95. Basar, T., Olsder, G.J.: *Dynamic Noncooperative Game Theory* (2nd ed.), Academic Press, New York, (1995)
- BP02. Benvenuto, N., Piazza, F.: On the complex backpropagation algorithm. *IEEE Trans. Sig. Proc.*, **40**(4), 967–969, (1992)
- BP02. Barahona, M., Pecora, L.M.: Synchronization in Small-World Systems. *Phys. Rev. Lett.* **89**, 054101, (2002)
- BP82. Barone, A., Paterno, G.: *Physics and Applications of the Josephson Effect*. New York: John Wiley & Sons, (1982)
- BP92. Benvenuto, N., Piazza, F.: On the complex backpropagation algorithm. *IEEE Trans. Sig. Proc.*, **40**(4), 967–969, (1992)
- BP96. Bassar, P.J., Pierpaoli, C.: Microstructural and physiological features of tissues elucidated by quantitative–diffusion–tensor MRI. *J. Magn. Reson. B*, **111**, 209–219, (1996)
- BPS75. Belavin, A.A., Polyakov, A.M., Swartz, A.S., Tyupkin, Yu.S.: SU(2) instantons discovered. *Phys. Lett. B* **59**, 85, (1975)
- BPS98. Blackmore, D.L., Prykarpatsky, Y.A., Samulyak, R.V.: The Integrability of Lie-invariant Geometric Objects Generated by Ideals in the Grassmann Algebra. *J. Nonlin. Math. Phys.*, **5**(1), 54–67, (1998)
- BPT94. Van den Broeck, C., Parrondo, J.M.R., Toral, R.: Noise-Induced Non-equilibrium Phase Transition. *Phys. Rev. Lett.* **73**, 3395–3398, (1994)
- BPT97. Van den Broeck, C., Parrondo, J.M.R., Toral, R., Kawai, R.: Non-equilibrium phase transitions induced by multiplicative noise. *Phys. Rev. E* **55**, 4084–4094, (1997)

- BPV03. Batista, A.M., de Pinto, S.E., Viana, R.L., Lopes, S.R.: Mode locking in small-world networks of coupled circle maps. *Physica A* **322**, 118, (2003)
- BR75. Bowen, R., Ruelle, D.: The ergodic theory of Axiom A flows. *Invent. Math.* **29**, 181-202, (1975)
- BR95. Bagnoli, F., Rechtman, R.: Chaos and Synchronization in Cellular Automata. In *Forecasting and Modeling for Chaotic and Stochastic Systems*, 145-154, A. Bellacicco, G. Koch, A. Vulpiani (ed), Collana Scientifica, Franco Angeli, Milan, Italy (1995)
- BR99. Bagnoli, F., Rechtman, R.: Synchronization and Maximum Lyapunov Exponent in Cellular Automata. *Phys. Rev. E* **59**, R1307, (1999)
- BRR95. Botina, J., Rabitz, H., Rahman, N.: Optimal control of chaotic Hamiltonian dynamics. *Phys. Rev. A* **51**, 923-933, (1995)
- BRT99. Bennati, E., Rosa-Clot, M., Taddei, S.: A Path Integral Approach to Derivative Security Pricing: I. Formalism and Analytical Results, *Int. Journ. Theor. Appl. Finance* **2**, 381, (1999)
- BS00. Becskei, A., Serrano, L.: Engineering stability in gene networks by autoregulation. *Nature*, **405**, 590-593, (2000)
- BS02. Bornholdt, S., Schuster, H.G. (eds): *Handbook of Graphs and Networks*. Wiley-VCH, Weinheim, (2002)
- BS70. Bhatia, N.P., Szego, G.P.: *Stability theory of dynamical systems*. Springer-Verlag, Heidelberg, (1979)
- BS73. Black, F., Scholes, M.: The Pricing of Options and Corporate Liabilities. *J. Polit. Econ.* **81**, 637-659, (1973)
- BS85. Bamber, D., van Santen, J.P.H.: How many parameters can a model have and still be testable? *J. Math. Psych.*, **29**, 443-473, (1985)
- BS91. Braitenberg, V., Shulz, A.: *Anatomy of the Cortex*. Springer, Berlin, (1991)
- BS93. Beck, C., Schlögl, F.: *Thermodynamics of chaotic systems*. Cambridge Univ. Press, Cambridge, (1993)
- BSM97. Breitenbach, G., Schiller, S., Mlynek, J.: Measurement of the quantum states of squeezed light. *Nature*, **387**, 471-475 (1997)
- BSV94. Bollé, D., Shim, G.M., Vinck, B., Zagrebnov, V.A.: Retrieval and chaos in extremely diluted Q-Ising neural networks. *J. Stat. Phys.* **74**, 565-582, (1994)
- BT00. Bellet, L.R., Thomas, L.E.: Asymptotic Behavior of Thermal Non-Equilibrium Steady States for a Driven Chain of Anharmonic Oscillators. *Commun. Math. Phys.* **215**, 1-24, (2000)
- BT03. Bauchau, O.A., Trainelli, L.: The Vectorial Parameterization of Rotation. *Non. Dyn.*, **31**(1), 71-92, (2003)
- BT93. Busemeyer, J.R., Townsend, J.T.: Decision field theory: A dynamic-cognitive approach to decision making in an uncertain environment. *Psych. Rev.*, **100**, 432-459, (1993)
- BW00. Barrat, A., Weigt, M.: On the properties of small-world network models. *Eur. Phys. J. B* **13**, 547, (2000)
- BZ02. Le Bihan, D., van Zijl, P.: From the diffusion coefficient to the diffusion tensor. *NMR Biomed.*, **15**(7-8), 431-4, (2002)

- Bab98. Babuška, R.: Fuzzy modeling: Principles, methods and applications. In C. Bonivento, C. Fantuzzi, and R. Rovatti, editors, *Fuzzy Logic Control: Advances in Methodology*, 187–220. World Scientific, Singapore, (1998)
- Bac96. T. Bäck, *Evolutionary Algorithm in Theory and Practice*. Oxford Press, (1996)
- Bae01. Baez, J.: The Meaning of Einstein's Equation. arXiv: gr-qc/0103044.
- Bae02. Baez, J.: Categorified gauge theory. Lecture in the Joint Spring Meeting of the Pacific Northwest Geometry Seminar and Cascade Topology Seminar, (2002)
- Bae97. Baez, J.: An introduction to n -categories. *7th Conference on Category Theory and Computer Science*, E. Moggi and G. Rosolini (eds), *Lecture Notes in Computer Science*, Springer, Berlin, (1997)
- Bae98. Von Baeyer, H.C.: All Shook Up. *The Sciences*, **38**(1), 12–14, (1998)
- Bai89. Bai-Iin, H.: *Elementary Symbolic Dynamics and Chaos in Dissipative Systems*, World Scientific, Singapore, (1989)
- Bal06. Balakrishnan, J.: A geometric framework for phase synchronization in coupled noisy nonlinear systems. *Phys. Rev. E* **73**, 036206, (2006)
- Bar93. Barry Jay, C.: *Matrices, Monads and the Fast Fourier Transform*. Univ. Tech., Sidney, (1993)
- Bau00. Baum, H.: Twistor and Killing spinors on Lorentzian manifolds and their relations to CR and Kaehler geometry. *Int. Congr. Diff. Geom. in memory of Alfred Gray*, Bilbao, Spain, (2000)
- Bax82. Baxter, R.: *Exactly Solved Models in Statistical Mechanics*. Academic Press, London, (1982)
- Bel81. Belbruno, E.A.: A new family of periodic orbits for the restricted problem, *Celestial Mech.*, **25**, 397–415, (1981)
- Ben67. Bénabou, J.: Introduction to bicategories. In: *Lecture Notes in Mathematics*. Springer, New York, (1967)
- Ben84. Benettin, G.: Power law behavior of Lyapunov exponents in some conservative dynamical systems. *Physica D* **13**, 211–213, (1984)
- Ben95. S. P. Benz: Superconductor-normal-superconductor junctions for programmable voltage standards. *Appl. Phys. Lett.* **67**, 2714–2716, (1995)
- Ber35. Bernstein, N.A.: *Investigations in Biodynamics of Locomotion* (in Russian). WIEM, Moscow, (1935)
- Ber47. Bernstein, N.A.: On the structure of motion (in Russian). Medgiz, Moscow, (1947)
- Ber84. Berry, M.V.: Quantal phase factor accompanying adiabatic change. *Proc. R. Soc. Lond. A* **392**, 45–57, (1984)
- Ber85. Berry, M.V.: Classical adiabatic angles and quantal adiabatic phase. *J. Phys. A* **18**, 15–27, (1985)

- Ber91. Bernier, O: Stochastic analysis of synchronous neural networks with asymmetric weights. *Europhys. Lett.* **16**, 531-536, (1991)
- Bie95. Bienenstock, E.: A model of neocortex. *Network: Computat. Neural Syst.* **6**, 179-224, (1995)
- Bih00. Le Bihan, D.: What to expect from MRI in the investigation of the central nervous system? *CR Acad. Sci. III*, **323**(4), 341-50, (2000)
- Bih03. Le Bihan, D.: Looking into the functional architecture of the brain with diffusion MRI. *Nature Rev. Neurosci.*, **4**(6), 469-80, (2003)
- Bih95. Le Bihan, D.: Diffusion and perfusion magnetic resonance imaging. Applications to functional MRI. Raven Press, New York (1995)
- Bih96. Le Bihan, D.: Functional MRI of the brain: principles, applications and limitations. *J. Neuroradiol.*, **23**(1), 1-5, (1996)
- Bil65. Billingsley, P.: *Ergodic Theory and Information*. Wiley, New York, (1965)
- Bir27. Birkhoff, G.D.: *Dynamical Systems*, Amer. Math. Soc., Providence, RI, (1927)
- Bla83. Blattner, R.: Nonlinear Partial Differential Operators and Quantization Procedure, in *Proceedings, Clausthall 1981*, Springer-Verlag, New York, 209-241, (1983)
- Bla90. Blahut, R.E.: *Principles and Practice of Information Theory*. Addison-Wesley, Reading, MA (1990)
- Blo66. Bloembergen, N.: *Nonlinear optics*. Benjamin, New York, (1965)
- Bob92. Bobbert, P.A.: Simulation of vortex motion in underdamped two-dimensional arrays of Josephson junctions. *Phys. Rev. B* **45**, 75407543, (1992)
- Boc04. Boccaro, N.: *Modeling complex systems*. Springer, Berlin, (2004)
- Bol01. Bollobás, B.: *Random Graphs*, (2nd ed). Cambridge Univ. Press, Cambridge, (2001)
- Bon95. Bontempi, G.: Modeling with uncertainty in continuous dynamical systems: the probability and possibility approach. IRIDIA – ULB Technical Report, 95-16, (1995)
- Bow70. Bowen, R.: Markov partitions for Axiom A diffeomorphisms. *Amer. J. Math.* **92**, 725-747 (1970)
- Bow73. Bowen, R.: Symbolic dynamics for hyperbolic flows. *Amer. J. Math.* **95**, 429-460, (1973)
- Bow75. Bowen, R.: Equilibrium states and the ergodic theory of Anosov diffeomorphisms. *Lecture Notes in Math.* **470**, Springer-Verlag, Berlin, (1975)
- Bre04. Brent, R.: A partnership between biology and engineering. *Nature*, **22**, 1211-1214, (2004)
- Bro01. Brockett, R.: *New Issues in the Mathematics of Control*. In *Mathematics Unlimited – 2001 and Beyond*, Springer, New York, (2001)

- Bro58. Broadbent, D.E.: Perception and communications. Pergamon Press, London, (1958)
- Bro86. Brooks, R.A.: A robust layered control system for a mobile robot. *IEEE Trans. Rob. Aut.* **2**(1), 14–23, (1986)
- Buz91. Buzan, T.: The Mind Map Book. Penguin, New York, (1991)
- CA93. Chialvo, D.R., Apkarian, V.: Modulated noisy biological dynamics: Three examples. *Journal of Statistical Physics*, **70**, 375–391, (1993)
- CA97. Chen, L., Aihara, K.: Chaos and asymptotical stability in discrete-time neural networks, *Physica D* **104**, 286–325, (1997)
- CAM05. Cvitanovic, P., Artuso, R., Mainieri, R., Tanner, G., Vattay, G.: Chaos: Classical and Quantum. ChaosBook.org, Niels Bohr Institute, Copenhagen, (2005)
- CC95. Christini, D.J., Collins, J.J.: Controlling Nonchaotic Neuronal Noise Using Chaos Control Techniques. *Phys. Rev. Lett.* **75**, 2782–2785, (1995)
- CC99. Cao, H.D., Chow, B.: Recent Developments on the Ricci Flow. *Bull. Amer. Math. Soc.* **36**, 59–74, (1999)
- CCC97. Caiani, L., Casetti, L., Clementi, C., Pettini, M.: Geometry of dynamics, Lyapunov exponents and phase transitions. *Phys. Rev. Lett.* **79**, 4361, (1997)
- CCC98. Caiani, L., Casetti, L., Clementi, C., Pettini, G., Pettini, M., Gatto, R.: Geometry of dynamics and phase transitions in classical lattice φ^4 theories *Phys. Rev. E* **57**, 3886, (1998)
- CCP96. Christiansen, F., Cvitanovic, P., Putkaradze, V.: Hopf’s last hope: spatio-temporal chaos in terms of unstable recurrent patterns. *Nonlinearity*, **10**, 1, (1997)
- CCP99. Casetti, L., Cohen, E.G.D., Pettini, M.: Origin of the Phase Transition in a Mean-Field Model. *Phys. Rev. Lett.*, **82**, 4160, (1999)
- CCV98. Castellanos, A., Coolen, A.C., Viana, L.: Finite-size effects in separable recurrent neural networks. *J. Phys. A: Math. Gen.* **31**, 6615–6634, (1998)
- CD98. Chen, G., Dong, X.: From Chaos to Order. Methodologies, Perspectives and Application. World Scientific, Singapore, (1998)
- CE91. Cvitanovic, P., Eckhardt, B.: Periodic orbit expansions for classical smooth flows. *J. Phys. A* **24**, L237, (1991)
- CF01. Cateau, H., Fukai, T.: Fokker-Planck approach to the pulse packet propagation in synfire chain. *Neu. Net.* **14**, 675–685, (2001)
- CF94. Crane, L., Frenkel, I.: Four dimensional topological quantum field theory, Hopf categories, and the canonical bases. *Jour. Math. Phys.* **35**, 5136–5154, (1994)
- CFL97. Crisanti, A., Falcioni, M., Lacorata, G., Purini, R., Vulpiani, A.: Characterization of a periodically driven chaotic dynamical system. *J. Phys. A, Math. Gen.* **30**, 371–383, (1997)

- CFP94. Crisanti, A., Falcioni, M., Paladin, G., Vulpiani, A.: Stochastic Resonance in Deterministic Chaotic Systems. *J. Phys. A* **27** L597, (1994)
- CG03. Carpenter, G.A., Grossberg, S.: Adaptive Resonance Theory. In M.A. Arbib (Ed.), *The Handbook of Brain Theory and Neural Networks*, Second Edition, Cambridge, MA: MIT Press, 87–90, (2003)
- CG83. Cohen, M.A., Grossberg, S.: Absolute stability of global pattern formation and parallel memory storage by competitive neural networks. *IEEE Trans. Syst., Man, Cybern.*, **13**(5), 815–826, (1983)
- CGP88. Cvitanovic, P., Gunaratne, G., Procaccia, I.: Topological and metric properties of Hénon-type strange attractors. *Phys. Rev. A* **38**, 1503–1520, (1988)
- CH03. Cohen, R., Havlin, S.: Scale-Free Networks Are Ultrasmall. *Phys. Rev. Lett.* **90**, 058701, (2003)
- CH03. De Carvalho, A., Hall, T.: Conjugacies between horseshoe braids. *Nonlinearity*, **16**, 1329–1338, (2003)
- CH64. Conway, E.D., Hopf, E.: Hamilton's Theory and Generalized Solutions of the Hamilton–Jacobi Equation. *J. Math. Mech.* **13**, 939–986, (1964)
- CK90. Crick, F., Koch, C.: Towards a neurobiological theory of consciousness. *Sem Neurosci* **2**, 263–275, (1990)
- CK97. Cutler, C.D., Kaplan, D.T.: (eds): *Nonlinear Dynamics and Time Series*. Fields Inst. Comm. **11**, American Mathematical Society, (1997)
- CKS05. Coolen, A.C.C., Kuehn, R., Sollich, P.: *Theory of Neural Information Processing Systems*. Oxford Univ. Press, Oxford, (2005)
- CL72. Craik, F., Lockhart, R.: Levels of processing: A framework for memory research. *J. Verb. Learn. & Verb. Behav.*, **11**, 671–684, (1972)
- CL81. Caldeira, A.O., Leggett, A.J.: Influence of Dissipation on Quantum Tunneling in Macroscopic Systems. *Phys. Rev. Lett.* **46**, 211, (1981)
- CL84. Cheng, T.-P., Li, L.-F.: *Gauge Theory of Elementary Particle Physics*. Clarendon Press, Oxford, (1984)
- CL99. Cassandras, C.G., Lafortune, S.: *Introduction to Discrete Event Systems*. Kluwer, Norwell, MA, (1999)
- CLP95. Casetti, L., Livi, R., Pettini, M.: Gaussian Model for Chaotic Instability of Hamiltonian Flows. *Phys. Rev. Lett.* **74**, 375378, (1995)
- CLV03. Chung, F., Lu, L. Vu, V.: The spectra of random graphs with given expected degrees. *Proc. Natl. Acad. Sci. U.S.A.* **100**(11), 6313–6318, (2003)
- CM97. Carpinteri, A., Mainardi, F.: *Fractals and Fractional Calculus in Continuum Mechanics*. Springer Verlag, Wien & New York, (1997)

- CMP98a. Claussen, J.C., Mausbach, T., Piel, A. Schuster, H.G.: Improved difference control of unknown unstable fixed-points: Drifting parameter conditions and delayed measurement. *Phys. Rev. E*, **58**(6), 7256–7260, (1998)
- CMP98b. Claussen, J.C., Mausbach, T., Piel, A. Schuster, H.G.: Memory difference control of unknown unstable fixed-points: Drifting parameter conditions and delayed measurement. *Phys. Rev. E* **58**(6), 7256–7260, (1998b)
- CN01. Cottrill-Shepherd, K., Naber M.: Fractional Differential Forms. *J. Math. Phys.* **42**, 2203-2212, (2001)
- CP02. Clementi, C. Pettini, M.: A geometric interpretation of integrable motions. *Celest. Mech. & Dyn. Astr.*, **84**, 263–281, (2002)
- CPC03. Casetti, L., Pettini, M., Cohen, E.G.D.: Phase transitions and topology changes in configuration space. *J. Stat. Phys.*, **111**, 1091, (2003)
- CPS93. Coolen, A.C., Penney, R.W., Sherrington, D.: Coupled dynamics of fast spins and slow interactions: An alternative perspective on replicas. *Phys. Rev. B* **48**, 1611616118, (1993)
- CQ69. Collins, A.M., Quillian, M.R.: Retrieval Time From Semantic Memory, *J. Verb. Learn. & Verb. Behav.*, **8**, 240–248, (1969)
- CR88. Cohen, E.G.D., Rondoni, L.: Note on phase-space contraction and entropy production in thermostatted Hamiltonian systems. *Chaos* **8**, 357-365, (1998)
- CRO04. Carinena, P., Regueiro, C., Otero, A. et al.: Landmark detection in mobile robotics using fuzzy temporal rules. *IEEE Trans. Fuz. Sys.* **12**(4), 423–435, (2004)
- CRV92. Celeghini, E., Rasetti, M., Vitiello, G.: Quantum Dissipation. *Annals Phys.*, **215**, 156, (1992)
- CS00. Cristianini, N., Shawe-Taylor, J.: Support Vector Machines. Cambridge Univ. Press, Cambridge, (2000)
- CS02. Claussen, J.C., Schuster, H.G.: Asymptotic Level Density of the Elastic Net Self-Organizing Feature Map. 939-944 in: J. R. Dorronsoro (Ed.): Artificial Neural Networks – ICANN’2002, Springer LNCS, Berlin, (2002)
- CS93. Coolen, A.C.C., Sherrington, D.: Dynamics of fully connected attractor neural networks near saturation. *Phys. Rev. Let.* **71** 3886, (1993)
- CS94. Coolen, A.C.C., Sherrington, D.: Order-parameter flow in the fully connected Hopfield model near saturation. *Phys. Rev. E* **49** 1921-934; and Erratum: Order-parameter flow in the fully connected Hopfield model near saturation, 5906, (1994)
- CS95. Chernikov, A.A., Schmidt, G.: Conditions for synchronization in Josephson-junction arrays. *Phys. Rev. E* **52**, 34153419, (1995).
- CS96. Constantine, G.M., Savits, T.H.: A multivariate Faà di Bruno formula with applications. *Trans. Amer. Math. Soc.* **348**(2), 503–520, (1996)

- CS98. Claussen, J.C., Schuster, H.G.: Stability borders of delayed measurement from time-discrete systems. arXiv nlin.CD/0204031, (1998)
- CT00. Cristianini, N., Taylor, J.S.: Introduction to Support Vector Machines. Cambridge Univ. Press, Cambridge, (2000)
- CT88. Cardoso, O., Tabelaing, P.: Anomalous diffusion in a linear array of vortices. Europhys. Lett. **7**(3), 225230, (1988)
- CV91. Cecotti, S., Vafa, C.: Topological – antitopological fusion, Nucl. Phys. B **367**, 359, (1991)
- CWM00. Clark, C.A., Werring, D.J., Miller, D.H.: Diffusion imaging of the spinal cord in vivo: estimation of the principal diffusivities and application to multiple sclerosis. Magn. Reson. Med. **43**, 133–138, (2000)
- CY05. Cao, Y., Ying, M.: Supervisory Control of Fuzzy Discrete Event Systems. IEEE Trans. Syst., Man, Cybern., B **35**, 366–371, (2005)
- CY89. Crutchfield, J.P., Young, K.: Computation at the onset of chaos. In Complexity, Entropy and the Physics of Information, 223, SFI Studies in the Sciences Complexity, Vol. VIII, W.H. Zurek (Ed.), Addison-Wesley, Reading, MA, (1989)
- CY89. Crutchfield, J.P., Young, K.: Computation at the onset of chaos. In *Complexity, Entropy and the Physics of Information*, p. 223, SFI Studies in the Sciences Complexity, Vol. VIII, W.H. Zurek (Ed.), Addison-Wesley, Reading, MA, (1989)
- Cal57. Calabi, E.: On Kähler manifolds with vanishing canonical class. In Algebraic geometry and topology: a symposium in honor of S. Lefschetz, eds. R.H. Fox, D.C. Spencer and A.W. Tucker, Princeton Univ. Press, (1957)
- Can86. Canarutto, D.: Bundle splittings, connections and locally principle fibred manifolds. Bull. U.M.I. Algebra e Geometria, Seria VI **V-D**, 18, (1986)
- Che52. Chernov, N.: A mesure of asymptotic efficiency for tests of hypothesis based on the sum of observations. Ann. Math. Stat. **23**, 493-507, (1952)
- Che55. Chevalley, C.: Theorie differential equations groupes de Lie. Vol. 1–3. Hermann D.C., Paris, (1955)
- Che96. Chern, S.S.: Riemannian Geometry as a Special Case of Finsler Geometry, Cont. Math., 51–58, Vol. 196, Amer. Math. Soc. Providence, RI, (1996)
- Chi79. Chirikov, B.V.: A universal instability of many-dimensional oscillator systems. Phys. Rep. **52**, 264-379, (1979)
- Chu94. Chua, L.O: Chua’s Circuit: an overview ten years later. J. Circ. Sys. Comp. **4**(2), 117-159, (1994)
- Cla02a. Claussen, J.C.: Generalized Winner Relaxing Kohonen Feature Maps. arXiv cond-mat/0208414, (2002)
- Cla02b. Claussen, J.C.: Floquet Stability Analysis of Ott-Grebogi-Yorke and Difference Control. arXiv:nlin.CD/0204060, (2002)
- Col02. Collins, P.: Symbolic dynamics from homoclinic tangles. Intern. J. Bifur. Chaos, **12**(3), 605–617, (2002)

- Col88. Coleman, S.: Aspects of Symmetry: Selected Erice Lectures. Cambridge Univ. Press, (1988)
- Coo01. Coolen, A.C.C.: Statistical Mechanics of Recurrent Neural Networks. In F. Moss, S. Gielen (eds.) Handbook of Biological Physics, **4**, Elsevier, (2001)
- Coo89. Cook, J.: The mean-field theory of a Q-state neural network model. J. Phys. A **22**, 2057, (1989)
- Cox92. Cox, E.: Fuzzy Fundamentals, IEEE Spectrum, 58–61, (1992)
- Cox94. Cox, E.: The Fuzzy Systems Handbook. AP Professional, (1994)
- Cra91. Crawford, J.: Clifford algebra: Notes on the spinor metric and Lorentz, Poincaré and conformal groups. J. Math. Phys. **32**, 576, (1991)
- Cri94. Crick, F.: The Astonishing Hypothesis. Charles Scribner's Sons, New York, (1994)
- Cvi00. Cvitanovic, P.: Chaotic field theory: a sketch. Physica A **288**, 61–80, (2000)
- Cvi91. Cvitanovic, P.: Periodic orbits as the skeleton of classical and quantum chaos. Physica, D **51**, 138, (1991)
- DB98. Dominguez, D.R.C., Bollé, D.: Self-control in sparsely coded networks. Phys. Rev. Lett. **80**, 2961-2964, (1998)
- DBO01. DeShazer, D.J., Breban, R., Ott, E., Roy, R.: Detecting Phase Synchronization in a Chaotic Laser Array. Phys. Rev. Lett. **87**, 044101, (2001)
- DCS98. Düring, A, Coolen, A.C., Sherrington, D.: Phase diagram and storage capacity of sequence processing neural networks. J. Phys. A: Math. Gen. **31**, 8607, (1998)
- DDP04. Doulamis, N., Doulamis, A., Panagakis, et al.: A combined fuzzy-neural network model for nonlinear prediction of 3D rendering workload in grid computing. IEEE Trans. Syst., Man, Cybern., B **34**, 1235-1247, (2004)
- DDT03. Daniels, B.C., Dissanayake, S.T.M., Trees, B.R.: Synchronization of coupled rotators: Josephson junction ladders and the locally coupled Kuramoto model. Phys. Rev. E **67**, 026216 (2003).
- DFM94. Dotsenko, V., Franz, S., Mézard, M.: Memorizing polymers' shapes and permutations. J. Phys. A: Math. Gen. **27**, 2351, (1994)
- DGA99. Diesmann, M., Gewaltig, M-O., Aertsen, A.D.: Stable propagation of synchronous spiking in cortical neural networks. Nature **402**, 529–533, (1999)
- DGO89a. Ding, M., Grebogi, C., Ott, E.: Evolution of attractors in quasi-periodically forced systems: From quasiperiodic to strange nonchaotic to chaotic. Phys. Rev. A **39**, 2593-2598, (1989)
- DGO89b. Ding, M., Grebogi, C., Ott, E.: The Dimension of Strange Nonchaotic Attractors. Phys. Lett. A **137**, 167-172, (1989)
- DGV95. Diekmann, O., van Gils, S.A., Verduyn-Lunel, S.M., Walther, H.-O.: Delay Equations. Appl. Math. Sci. **110**, Springer-Verlag, New York, (1995)

- DGY97. Ding, M., Grebogi, C., Yorke, J.A.: Chaotic dynamics. In *The Impact of Chaos on Science and Society*, Eds. C. Grebogi and J.A. Yorke, 1-17, United Nations Univ. Press, Tokyo, (1997)
- DGZ87. Derrida, B., Gardner, E., Zippelius, A.: An exactly solvable asymmetric neural network model. *Europhys. Lett.* **4**, 167, (1987)
- DH85. Douady, A., Hubbard, J.: On the dynamics of polynomial-like mappings. *Ann. scient. Éc. Norm. Sup.*, 4^e ser. **18**, 287–343, (1985)
- DH90. Ding, M.-Z., Hao, B.-L.: Systematics of the periodic windows in the Lorenz model and its relation with the antisymmetric cubic map. *Commun. Theor. Phys.* **9**, 375, (1988)
- DHR00. Dauxois, T., Holdsworth, P., Ruffo, S.: Violation of ensemble equivalence in the antiferromagnetic mean-field XY model. *Eur. Phys. J. B* **16**, 659, (2000)
- DHS91. Domany, E., van Hemmen, J.L., Schulten, K. (eds.): *Models of Neural Networks*. Springer, Berlin, (1991)
- DK87. Dress, W.B., Knisley, J.R.: A Darwinian approach to artificial neural systems. *Proc IEEE Conf. Sys., Man Cybern.*, (1987)
- DL01. Dasgupta, A., Loll, R.: A proper-time cure for the conformal sickness in quantum gravity. *Nucl. Phys. B* **606**, 357–379, (2001)
- DM03. Dorogovtsev, S.N., Mendes, J.F.F.: *Evolution of Networks*. Oxford Univ. Press, (2003)
- DN79. Devaney, R., Nitecki, Z.: Shift automorphisms in the Hénon mapping. *Comm. math. Phys.* **67**, 137–48, (1979)
- DP80. Dubois, D., Prade, H.: *Fuzzy Sets and Systems*. Academic Press, New York, (1980)
- DR94. Dittrich, W., Reuter, M.: *Classical and Quantum Dynamics*. Springer Verlag, Berlin, (1994)
- DS74. Davey, A., Stewartson, K.: On Three-Dimensional Packets of Surface Waves. *Proc. R. Soc. A* **338**, 101-110, (1974)
- DSS90. Ditto, W.L., Spano, M.L., Savage, H.T., Rauseo, S.N., Heagy, J., Ott, E.: Experimental observation of a strange nonchaotic attractor. *Phys. Rev. Lett.* **65**, 533-536, (1990)
- DSS96. Dote, Y., Strefezza, M., and Suitno, A.: Neuro fuzzy robust controllers for drive systems. in *Neural Networks Applications*, ed. P.K. Simpson, IEEE Tec. Upd. Ser., New York, (1996)
- DT89. Ditzinger, T., Haken, H.: Oscillations in the perception of ambiguous patterns: A model based on synergetics. *Biological Cybernetics*, **61**, 279–287, (1989)
- DT90. Ditzinger, T., Haken, H.: The impact of fluctuations on the recognition of ambiguous patterns. *Biol. Cyber.* **63**, 453–456, (1990)
- DT95. Denniston, C., Tang, C.: Phases of Josephson Junction Ladders. *Phys. Rev. Lett.* **75**, 3930, (1995)
- DTP02. Dauxois, T., Theodorakopoulos, N., Peyrard, M.: Thermodynamic instabilities in one dimension: correlations, scaling and solitons. *J. Stat. Phys.* **107**, 869, (2002)

- DTP91. Douek, P., Turner, R., Pekar, J., Patronas, N.J., Le Bihan, D.: MR color mapping of myelin fiber orientation. *J. Comput. Assist. Tomogr.*, **15**, 923–929, (1991)
- DW87. Durbin, R., Willshaw, D.: An analogue approach to the Travelling Salesman Problem using an Elastic Net Method. *Nature* **326**, 689–691, (1987)
- DW95. Dormayer, P., Lani-Wayda, B.: Floquet multipliers and secondary bifurcations in functional differential equations: Numerical and analytical results, *Z. Angew. Math. Phys.* **46**, 823–858, (1995)
- Dai87. Daido, H.: Population Dynamics of Randomly Interacting Self-Oscillators. *Prog. Theor. Phys.* **77**, 622–634, (1987)
- Dai92. Daido, H.: Quasientrainment and slow relaxation in a population of oscillators with random and frustrated interactions. *Phys. Rev. Lett.*, **68**, 1073–1076, (1992)
- Das02. Dasgupta, A.: The real Wick rotations in quantum gravity. *JHEP*, **0207**, (2002)
- Dav81. Davydov, A.S.: *Biology and Quantum Mechanics*, Pergamon Press, New York, (1981)
- Dav87. Davis, L. (ed): *Genetic Algorithms and Simulated Annealing*. Pitman. London, (1987)
- Dav89. Davies, E.B.: *Heat Kernels and Spectral Theory*. Cambridge Univ. Press, (1989)
- Dav91. Davydov, A.S.: *Solitons in Molecular Systems*. (2nd ed), Kluwer, Dordrecht, Ger, (1991)
- Dev89. Devaney, R.: *An Introduction to Chaotic Dynamical Systems*. Addison Wesley Publ. Co. Reading MA, (1989)
- Die69. Dieudonne, J.A.: *Foundations of Modern Analysis* (in four volumes). Academic Press, New York, (1969)
- Die88. Dieudonne, J.A.: *A History of Algebraic and Differential Topology 1900–1960*. Birkhäuser, Basel, (1988)
- Dir82. Dirac, P.A.M.: *Principles of Quantum Mechanics*. (4th ed.), Oxford Univ. Press, (1982)
- Dom78. de Dominicis, C.: Dynamics as a substitute for replicas in systems with quenched random impurities. *Phys. Rev. B* **18**, 4913–4919, (1978)
- Dor94. Dorcak, L.: *Numerical Models for Simulation the Fractional-Order Control Systems*, UEF SAV, The Academy of Sciences, Inst. of Exper. Phys., Kosice, Slovak Republic, (1994)
- Dow02. Dowker, F.: Topology change in quantum gravity. In *Proceedings of Stephen Hawking's 60th birthday conference*, Cambridge, UK, (2002)
- Dri77. Driver, R.D.: *Ordinary and delay differential equations*. Applied Mathematical Sciences 20, Springer Verlag, New York, (1977)
- Dus84. Dustin, P.: *Microtubules*. Springer, Berlin, (1984)
- EB01. Endy, D., Brent, R.: Modeling cellular behavior. *Nature* **409**, 391–395, (2001)
- EIS67. Eccles, J.C., Ito M., Szentagothai J.: *The Cerebellum as a Neuronal Machine*. Springer, Berlin, (1967)

- EL00. Elowitz, M.B., Leibler, S.: A synthetic oscillatory network of transcriptional regulators. *Nature*, **403**, 335–338, (2000)
- EL78. Eells, J., Lemaire, L.: A report on harmonic maps. *Bull. London Math. Soc.* **10**, 1–68, (1978)
- EM90. Evans, D.J., Morriss, G.P.: *Statistical mechanics of non-equilibrium fluids*. Academic Press, New York, (1990)
- EMN92. Ellis, J., Mavromatos, N., Nanopoulos, D.V.: String theory modifies quantum mechanics. CERN-TH/6595, (1992)
- EMN99. Ellis, J., Mavromatos, N., Nanopoulos, D.V.: A microscopic Liouville arrow of time. *Chaos, Solit. Fract.*, **10**(2–3), 345–363, (1999)
- EP98. Ershov, S.V., Potapov, A.B.: On the concept of Stationary Lyapunov basis. *Physica D*, **118**, 167, (1998)
- EPG95. Ernst, U., Pawelzik, K., Geisel, T.: Synchronization induced by temporal delays in pulse-coupled oscillators. *Phys. Rev. Lett.* **74**, 1570, (1995)
- EPR99. Eckmann, J.P., Pillet, C.A., Rey-Bellet, L.: Non-equilibrium statistical mechanics of anharmonic chains coupled to two heat baths at different temperatures. *Commun. Math. Phys.*, **201**, 657–697, (1999)
- ER85. Eckmann, J.P., Ruelle, D.: Ergodic theory of chaos and strange attractors, *Rev. Mod. Phys.*, **57**, 617–630, (1985)
- ESH98. Elson, R.C., Selverston, A.I., Huerta, R. *et al.*: Synchronous Behavior of Two Coupled Biological Neurons. *Phys. Rev. Lett.* **81**, 5692–5695, (1998)
- EV87. Ehresmann, A.C., Vanbremeersch, J.-P.: Hierarchical Evolutionary Systems: A Mathematical Model for Complex Systems. *Bul. Math. Bio.* **49**(1), 13–50, (1987)
- EV96. Ehresmann, A.C., Vanbremeersch, J.-P.: Multiplicity Principle and emergence in MES. *Journ. Syst. Analysis, Modeling, Simulation* **26**, 81–117, (1996)
- Ecc64. Eccles, J.C.: *The Physiology of Synapses*. Springer, Berlin, (1964)
- Elk99. Elkin, V.I.: *Reduction of Nonlinear Control Systems. A Differential Geometric Approach*, Kluwer, Dordrecht, (1999)
- Elm90. Elman, J.: Finding structure in time. *Cogn. Sci.* **14**, 179–211, (1990)
- Elw82. Elworthy, K.D.: *Stochastic Differential Equations on Manifolds*. Cambridge Univ. Press, Cambridge, (1982)
- Ens05. Enss, C. (ed): *Cryogenic Particle Detection. Topics in Applied Physics* **99**, Springer, New York, (2005)
- Erm81. Ermentrout, G.B.: The behavior of rings of coupled oscillators. *J. Math. Biol.* **12**, 327, (1981)
- Erm96. Ermentrout, G.B.: Type I membranes, phase resetting curves, and synchrony. *Neural Computation*, **8**(5), 979–1001, (1996)
- FCS99. Franzosi, R., Casetti, L., Spinelli, L., Pettini, M.: Topological aspects of geometrical signatures of phase transitions. *Phys. Rev. E*, **60**, 5009–5012, (1999)
- FD94. Feldman, D.E., Dotsenko, V.S.: Partially annealed neural networks. *J. Phys. A: Math. Gen.* **27**, 4401–4411, (1994)

- FFF94. Fatibene, L., Ferraris, M., Francaviglia, M.: Nöther formalism for conserved quantities in classical gauge field theories II. *J. Math. Phys.*, **35**, 1644, (1994)
- FGR98. Fajstrup, L., Goubault, E., Raussen, M.: Detecting Deadlocks in Concurrent Systems, In D. Sangiorgi R. de Simone, editor, *CONCUR '98; Concurrency Theory*, number 1466 in *Lecture Notes in Computer Science*, 332–347. Springer, Berlin, (1998)
- FH65. Feynman, R.P., Hibbs, A.R.: *Quantum Mechanics and Path Integrals*, McGraw-Hill, New York, (1965)
- FH96. Fang, H.-P., Hao, B.-L.: Symbolic dynamics of the Lorenz equations. *Chaos, Solitons, and Fractals*, **7**, 217–246, (1996)
- FHI00. Fujii, K., Hayashi, D., Inomoto, O., Kai, S: Noise-Induced Entrainment between Two Coupled Chemical Oscillators in Belousov-Zhabotinsky Reactions. *Forma* **15**, 219, (2000)
- FK39. Frenkel, J., Kontorova, T.: On the theory of plastic deformation and twinning. *Phys. Z. Sowiet Union*, **1**, 137–49, (1939)
- FK82. Ferraris, M., Kijowski, J.: On the equivalence of the relativistic theories of gravitation. *GRG* **14**, 165, (1982)
- FK83. Frölich, H., Kremer, F.: *Coherent Excitations in Biological Systems*. Springer, New York, (1983)
- FK88. Fontanari, J.F., Köberle, R.: Information Processing in Synchronous Neural Networks. *J. Physique* **49**, 13, (1988)
- FKP95. Feudel, U., Kurths, J., Pikovsky, A.S.: Strange Nonchaotic Attractor in a Quasiperiodically Forced Circle Map. *Physica D* **88**, 176–186, (1995)
- FK01. Fonarev, A., Kryzhanovsky, B.V.: Parametric dynamic neural network recognition power. *Opt. Mem. Neu. Net.* **10**(4), 31–48, (2001)
- FMV01. Festa, R., Mazzino, A., Vincenzi, D.: An analytical approach to chaos in Lorenz-like systems. A class of dynamical equations. *Europhys. Lett.* **56**, 47–53, (2001)
- FMV02. Festa, R., Mazzino, A., Vincenzi, D.: Lorenz-like systems and classical dynamical equations with memory forcing: An alternate point of view for singling out the origin of chaos. *Phys. Rev. E* **65**, 046205, (2002)
- FP04. Franzosi, R., Pettini, M.: Theorem on the origin of phase transitions. *Phys. Rev. Lett.*, **92**(6), 60601, (2004)
- FPS00. Franzosi, R., Pettini, M., Spinelli, L.: Topology and phase transitions: a paradigmatic evidence. *Phys. Rev. Lett.* **84**(13), 2774–2777, (2000)
- FPU55. Fermi, E., Pasta, J.R., Ulam, S.M.: *Studies of nonlinear problems*, Los Alamos Scientific Laboratory Report No. LA-1940, (1955)
- FR04. Forger, M., Römer, H.: Currents and the energy-momentum tensor in classical field theory: a fresh look at an old problem. *Ann. Phys. (N.Y.)* **309**, 306–389, (2004)
- FS92. Freeman, J.A., Skapura, D.M.: *Neural Networks: Algorithms, Applications, and Programming Techniques*. Addison-Wesley, Reading, MA, (1992)

- FT97. Fukai, T., Tanaka, S.: A simple neural network exhibiting selective activation of neuronal ensembles: from winner-take-all to winners-share-all. *Neural Comp.* **9**, 77-97, (1997)
- FY83. Fujisaka, H., Yamada, T.: Amplitude Equation of Higher-Dimensional Nikolaevskii Turbulence. *Prog. Theor. Phys.* **69**, 32 (1983)
- Fed69. Federer, H.: *Geometric Measure Theory*. Springer, New York, (1969)
- FW95. Filatrella, G., Wiesenfeld, K.: Magnetic-field effect in a two-dimensional array of short Josephson junctions. *J. Appl. Phys.* **78**, 1878-1883, (1995)
- Fei78. Feigenbaum, M.J.: Quantitative universality for a class of nonlinear transformations. *J. Stat. Phys.* **19**, 25-52, (1978)
- Fei79. Feigenbaum, M.J.: The universal metric properties of nonlinear transformations. *J. Stat. Phys.* **21**, 669-706, (1979)
- Fen02. Feng, G.: An Approach to Adaptive Control of Fuzzy Dynamic Systems. *IEEE Trans. Fuzzy Systt.* **10**(2), 268-275, (2002)
- Fer98. Fersht, A.: *Structure and Mechanism in Protein Science*. W.H. Freeman, New York, (1998)
- Fer99. Ferber, J.: *Multi-Agent Systems. An Introduction to Distributed Artificial Intelligence*. Addison-Wesley, Reading, MA, (1999)
- Fey48. Feynman, R.P.: Space-time Approach to Nonrelativistic Quantum Mechanics. *Rev. Mod. Phys.* **20**, 367-387, (1948)
- Fey51. Feynman, R.P.: An Operator Calculus Having Applications in Quantum Electrodynamics. *Phys. Rev.* **84**, 108-128, (1951)
- Fey72. Feynman, R.P.: *Statistical Mechanics, A Set of Lectures*. WA Benjamin, Inc., Reading, MA, (1972)
- Fey98. Feynman, R.P.: *Quantum Electrodynamics. Advanced Book Classics*, Perseus Publishing, (1998)
- Fit61. FitzHugh, R.A.: Impulses and physiological states in theoretical models of nerve membrane. *Biophys J.*, **1**, 445-466, (1961)
- Fla63. Flanders, H.: *Differential Forms with Applications to the Physical Sciences*. Acad. Press, (1963)
- Fog96. Fogel, D.: *Evolutionary Computation*. IEEE Press, (1996)
- For90. Fordy, A.P. (ed.): *Soliton Theory: A Survey of Results*. MUP, Manchester, UK, (1990)
- Fri88. Friberg, O.: A Set of Parameters for Finite Rotations and Translations. *Comp. Met. App. Mech. Eng.*, **66**, 163-171, (1988)
- GBL03. Gardner, T.S., di Bernardo, D., Lorenz, D., Collins, J.J.: Inferring genetic networks and identifying compound mode of action via expression profiling. *Science*, **301**, 102-105, (2003)
- GC00. Gollub, J.P., Cross, M.C.: Many systems in nature are chaotic in both space and in time. *Nature*, **404**, 710-711, (2000)
- GC95a. Gallavotti, G., Cohen, E.G.D.: Dynamical ensembles in non-equilibrium statistical mechanics. *Phys. Rev. Letters* **74**, 2694-2697, (1995)
- GC95b. Gallavotti, G., Cohen, E.G.D.: Dynamical ensembles in stationary states. *J. Stat. Phys.* **80**, 931-970, (1995)

- GDL00. Guardiola, X., Diaz-Guilera, A., Llas, M., Perez, C.J.: Synchronization, diversity, and topology of networks of integrate and fire oscillators. *Phys. Rev. E* **62**, 5565, (2000)
- GDM87. Gardner, E., Derrida, B., Mottishaw, P.: Zero temperature parallel dynamics for infinite range spin glasses and neural networks. *J. de Physique*. **48**, 741, (1987)
- GEH02. Guet, C., Elowitz, M.B., Hsing, W., Leibler, S.: Combinatorial synthesis of genetic networks. *Science*, **296**(5572), 1466-1470, (2002)
- GF02. Grantner, J.L., Fodor, G.A.: Fuzzy automaton for intelligent hybrid control systems. In *Proc. 2002 IEEE Inter. Conf. Fuzzy Syst.*, 1027-1032, (2002)
- GG03. Gaucher, P., Goubault, E.: Topological Deformation of Higher Dimensional Automata. *Homology, Homotopy and Applications*, **5**(2), 39–82, (2003)
- GGs81. Guevara, M.R., Glass, L., Shrier, A.: Phase locking, period-doubling bifurcations, and irregular dynamics in periodically stimulated cardiac cells. *Science* **214**, 1350-53, (1981)
- GH00. Gade, P.M., Hu, C.K.: Synchronous chaos in coupled map lattices with small-world interactions. *Phys. Rev. E* **62**, 64096413, (2000)
- GH83. Guckenheimer, J., Holmes, P.: *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*. Springer-Verlag, Berlin, (1983)
- GJ94. van Groesen E., De Jager E.M.: (Editors) *Mathematical Structures in Continuous Dynamical Systems. Studies in Mathematical Physics*, vol. 6 North-Holland, Amsterdam, (1994)
- GK92. Georgiou, G.M., Koutsougeras, C.: Complex domain back-propagation. *Anal. Dig. Sig. Proc.*, **39**(5), 330–334, (1992)
- GKE89. Gray, C.M., König, P., Engel, A.K., Singer, W.: Oscillatory responses in cat visual cortex exhibit inter-columnar synchronization which reflects global stimulus properties. *Nature*, **338**, 334-337, (1989)
- GKM99. Garg, V.K., Kumar, R., Marcus, S.I.: A probabilistic language formalism for stochastic discrete event systems. *IEEE Trans. Automat. Contr.*, **44**, 280-293, (1999)
- GM04. Grinza, P., Mossa, A.: Topological origin of the phase transition in a model of DNA denaturation. *Phys. Rev. Lett.* **92**(15), 158102, (2004)
- GM84. De Groot, S.R., Mazur, P.: *Nonequilibrium thermodynamics*. Dover, New York, (1984)
- GM88. Gowitzke, B.A., Milner, M.: *Scientific Bases of Human Movement*. Williams and Wilkins, Baltimore, (1988)
- GMK94. Gabbiani, F., Midtgaard, J., and Knoepfl, T.: Synaptic integration in a model of cerebellar granule cells. *J. Neurophysiol.*, **72**, 999–1009, (1994)
- GMP05. Grunwald, P., Myung, I.J., Pitt, M.A.(eds.): *Advances in Minimum Description Length: Theory and Applications*. MIT Press, Cambridge, MA (2005)

- GN90. Gaspard, P., Nicolis, G.: Transport properties, Lyapunov exponents, and entropy per unit time. *Phys. Rev. Lett.* **65**, 1693-1696, (1990)
- GNR91. Gross, P., Neuhauser, D., Rabitz, H.: Optimal Control of Unimolecular Reactions in the Collisional Regime. *J. Chem. Phys.* **94**, 1158, (1991)
- GNR92. Gross, P., Neuhauser, D., Rabitz, H.: Optimal control of curve-crossing systems. *J. Chem. Phys.* **96**, 2834-2845, (1992)
- GNR93. Gross, P., Neuhauser, D., Rabitz, H.: Teaching lasers to control molecules in the presence of laboratory field uncertainty and measurement imprecision. *J. Chem. Phys.* **98**, 4557-4566, (1993)
- GOP84. Grebogi, C., Ott, E., Pelikan, S., Yorke, J.A.: Strange attractors that are not chaotic. *Physica D* **3**, 261-268, (1984)
- GOT99. Giles, C.L., Omlin, C.W., Thornber, K.K.: Equivalence in knowledge representation: Automata, recurrent neural networks and dynamical fuzzy systems. *Proc. IEEE*, **87**(9), 1623-1640, (1999)
- GOY87. Grebogi, C., Ott, E., Yorke, J.A.: Chaos, strange attractors, and fractal basin boundaries in nonlinear dynamics. *Science*, **238**, 632-637, (1987)
- GP83a. Grassberger, P., Procaccia, I.: Measuring the Strangeness of Strange Attractors. *Phys. D* **9**, 189-208, (1983)
- GP83b. Grassberger, P., Procaccia, I.: Characterization of Strange Attractors. *Phys. Rev. Lett.* **50**, 346-349, (1983)
- GPS95. Gordon, R., Power, A.J., Street, R.: Coherence for tricatagories. *Memoirs Amer. Math. Soc.* **117**(558), (1995)
- GRS96. Gilks, W.R., Richardson, S., Spiegelhalter, D.J.: *Markov Chain Monte Carlo in Practice*. Chapman & Hall, (1996)
- GS98. Grosche, C., Steiner, F.: *Handbook of Feynman path integrals*. Springer Tracts in Modern Physics 145, Springer, Berlin, (1998)
- GSD92. Garfinkel, A., Spano, M.L., Ditto, W.L., Weiss, J.N.: Controlling cardiac chaos. *Science* **257**, 1230-1235, (1992)
- GSS03. Godzik, N., Schoenauer, M., Sebag, M.: Evolving symbolic controllers. In G.R. et al. (ed.) *Applications of Evolutionary Computing*, *Lecture Notes in Computer Science* **2611**, 638-650, (2003)
- GT77. Grossmann, S., Thomae, S.: Invariant distributions and stationary correlation functions of one-dimensional discrete processes. *Z. Naturforsch. A* **32**, 1353-1363, (1977)
- GVP94. van Gelderen, P., de Vleeschouwer, M.H.M., des Pres, D., et al.: Water diffusion and acute stroke. *Magn. Reson. Med.* **31**, 154-163, (1994)
- GWA01. Grewal, M.S., Weill, L.R., Andrews, A.P.: *Global Positioning Systems, Inertial Navigation, and Integration*. Wiley, New York, (2001)
- GZ94. Glass, L., Zeng, W.: Bifurcations in flat-topped maps and the control of cardiac chaos. *Int. J. Bif. Chaos* **4**, 1061-1067, (1994)

- Gal83. Gallavotti, G.: *The Elements of Mechanics*. Springer-Verlag, Berlin, (1983)
- Gar85. Gardiner, C.W.: *Handbook of Stochastic Methods for Physics, Chemistry and Natural Sciences*, (2nd ed). Springer-Verlag, New York, (1985)
- Gar94. Gardiner, C.W.: *Handbook of Stochastic Methods*. Springer, Berlin, (1994)
- GGK67. Gardner, C.S., Greene, J.M., Kruskal, M.D., Miura, R.M.: Method for solving the Korteweg-de Vries equation. *Phys. Rev. Let.*, **19**, 1095–97, (1967)
- Gau02a. Gaucher, P.: Homotopy invariants of higher-dimensional categories and concurrency in computer science. *Mathematical Structure in Computer Science* **10**(4), 481–524, (2000)
- Gau02b. Gaucher, P.: From Concurrency to Algebraic Topology. *Electronic Notes in Theoretical Computer Science* **39**(2), 19, (2000)
- Gau02c. Gaucher, P.: About the globular homology of higher-dimensional automata. *Cahiers de Topologie et Géométrie Diff*
- Gau03. Gaucher, P.: The branching nerve of HDA and the Kan condition. *Theo. Appl. Categ.* **11**(3), 75–106, (2003)
- Geo88. Georgii, H.O.: *Gibbs Measures and Phase Transitions*. Walter de Gruyter, Berlin, (1988)
- Ghe90. Ghez, C.: Introduction to motor system. In: Kandel, E.K. and Schwarz, J.H. (eds.) *Principles of neural science*. 2nd ed. Elsevier, Amsterdam, 429–442, (1990)
- Ghe91. Ghez, C.: Muscles: Effectors of the Motor Systems. In: *Principles of Neural Science*. 3rd Ed. (Eds. E.R. Kandel, J.H. Schwartz, T.M. Jessell), Appleton and Lange, Elsevier, 548–563, (1991)
- Gil81. Gilmor, R.: *Catastrophe Theory for Scientists and Engineers*. Wiley, New York, (1981)
- Gla63a. Glauber, R.J.: The Quantum Theory of Optical Coherence. *Phys. Rev.* **130**, 2529–2539, (1963)
- Gla63b. Glauber, R.J.: Coherent and Incoherent States of the Radiation Field. *Phys. Rev.* **131**, 2766–2788, (1963)
- Gle87. Gleick, J.: *Chaos: Making a New Science*. Penguin–Viking, New York, (1987)
- Gol88. Goldberg, D.: *Genetic Algorithms*, Addison-Wesley, Reading, MA, (1988)
- Gol92. Goldenfeld, N.: *Lectures on Phase Transitions and the Renormalization Group*. Addison-Wesley, Reading, (1992)
- Gol99. Goldberger, A.L.: Nonlinear Dynamics, Fractals, and Chaos Theory: Implications for Neuroautonomic Heart Rate Control in Health and Disease. In: Bolis CL, Licinio J, eds. *The Autonomic Nervous System*. World Health Organization, Geneva, (1999)
- Gol99. Gold, M.: *A Kurt Lewin Reader, the Complete Social Scientist*. Am. Psych. Assoc., Washington, (1999)

- Gom94. Gómez, J.C.: Using symbolic computation for the computer aided design of nonlinear (adaptive) control systems. Tech. Rep. EE9454, Dept. Electr. and Comput. Eng., Univ. Newcastle, Callaghan, NSW, AUS, (1994)
- Goo98. Goodwine, J.W.: Control of Stratified Systems with Robotic Applications. PhD thesis, California Institute of Technology, Pasadena, Cal, (1998)
- Got96. Gottlieb, H.P.W.: Question #38. What is the simplest jerk function that gives chaos? *Am. J. Phys.*, **64**(5), 525, (1996)
- Gou95. Goubault, E.: The Geometry of Concurrency. PhD thesis, Ecole Normale Supérieure, (1995)
- Goz83. Gozzi, E.: Functional-integral approach to Parisi-Wu stochastic quantization: Scalar theory. *Phys. Rev. D* **28**, 1922, (1983)
- Gra90. Granato, E.: Phase transitions in Josephson-junction ladders in a magnetic field. *Phys. Rev. B* **42**, 4797-4799, (1990)
- Gre79. Greene, J.M.: A method for determining a stochastic transition. *J. Math. Phys.* **20**, 1183-1201, (1979)
- Gri83a. Griffiths, P.A.: Exterior Differential Systems and the Calculus of Variations, Birkhauser, Boston, (1983)
- Gri83b. Griffiths, P.A.: Infinitesimal variations of Hodge structure. III. Determinantal varieties and the infinitesimal invariant of normal functions. *Compositio Math.*, **50**(2-3), 267-324, (1983)
- Gro69. Grossberg, S.: Embedding fields: A theory of learning with physiological implications. *J. Math. Psych.* **6**, 209-239, (1969)
- Gro82. Grossberg, S.: Studies of Mind and Brain. Dordrecht, Holland, (1982)
- Gro83. Grothendieck, A.: Pursuing stacks (Unpublished manuscript, distributed from UCNW). Bangor, UK, (1983)
- Gro88. Grossberg, S.: Neural Networks and Natural Intelligence. MIT Press, Cambridge, MA, (1988)
- Gro99. Grossberg, S.: How does the cerebral cortex work? Learning, attention and grouping by the laminar circuits of visual cortex. *Spatial Vision* **12**, 163-186, (1999)
- Gro05. Gros, C.: Self-Sustained Thought Processes in a Dense Associative Network. Springer Lecture Notes in AI **3698**, 375-388, (2005)
- Gru00. Grunwald, P.: The minimum description length principle. *J. Math. Psych.*, **44**, 133-152, (2000)
- Gru99. Grunwald, P.: Viewing all models as 'probabilistic'. Proceedings of the Twelfth Annual Conference on Computational Learning Theory (COLT' 99), Santa Cruz, CA, (1999)
- Gun03. Gunion, J.F.: Class Notes on Path-Integral Methods. U.C. Davis, 230B, (2003)
- Gut90. Gutzwiller, M.C.: Chaos in Classical and Quantum Mechanics, Springer, New York, (1990)
- Gut98. Gutkin, B.S., Ermentrout, B.: Dynamics of membrane excitability determine interspike interval variability: A link between spike generation mechanisms and cortical spike train statistics. *Neural Comput.*, **10**(5), 1047-1065, (1998)

- HB87. Havlin, S., Ben-Avraham, D.: Diffusion in disordered media. *Adv. Phys.* **36**, 695-798, (1987)
- HBB96. Houk, J.C., Buckingham, J.T., Barto, A.G.: Models of the cerebellum and motor learning. *Behavioral and Brain Sciences*, **19**(3), 368-383, (1996)
- HCK02a. Hong, H., Choi, M.Y., Kim, B.J.: Synchronization on small-world networks. *Phys. Rev. E* **65**, 26139, (2002)
- HCK02b. Hong, H., Choi, M.Y., Kim, B.J.: Phase ordering on small-world networks with nearest-neighbor edges. *Phys. Rev. E* **65**, 047104, (2002)
- HCT97. Hall, K., Christini, D.J., Tremblay, M., Collins, J.J., Glass, L., Billette, J.: Dynamic Control of Cardiac Alternans. *Phys. Rev. Lett.* **78**, 4518-4521, (1997)
- HD89. Horn, D., Usher, M.: Neural networks with dynamical thresholds. *Phys. Rev. A* **40**, 1036-1044, (1989)
- HDR02. Hasty, J., Dolnik, M., Rottschäfer, V., Collins, J.J.: A synthetic gene network for entraining and amplifying cellular oscillations. *Phys. Rev. Lett.* **88**(14), 1-4, (2002)
- HE79. Hawking, S.W., Israel, W. (ed.): General relativity: an Einstein centenary survey. Cambridge Univ. Press, Cambridge, (1979)
- HG02. Hojati, M., Gazor, S.: Hybrid adaptive fuzzy identification and control of nonlinear systems. *IEEE Trans. Fuz. Sys.* **10**(2), 211-221, (2002)
- HH-N88. Haines, K., Hecht-Nielsen, R.: A BAM with Increased Information Storage Capacity. In *Proc. IEEE Conf. Neural Networks*, Vol 1, 181-190, San Diego, (1988)
- HH52. Hodgkin, A.L., Huxley, A.F.: A quantitative description of membrane current and application to conduction and excitation in nerve. *J. Physiol.*, **117**, 500-544, (1952)
- HH94. Heagy, J.F., Hammel, S.M.: The birth of strange nonchaotic attractors. *Physica D* **70**, 140-153, (1994)
- HH95. Hopfield, J.J., Hertz, A.V.M.: Rapid local synchronization of action potentials: Toward computation with coupled integrate-and-fire neurons. *Proc. Natl. Acad. Sci. USA*, **92**, 6655-6662, (1995)
- HH97. Harmon, M.E., Harmon, S.S.: Reinforcement learning: A tutorial. *Tec. Rep. Wright Laboratory and Wright State University*, (1997)
- HHP95. Herrmann, M., Hertz, J., Prügel-Bennett, A.: Analysis of syn-fire chains. *Network: Computat. Neural Syst.* **6**, 403-414, (1995)
- HI97. Hoppensteadt, F.C., Izhikevich, E.M.: *Weakly Connected Neural Networks*. Springer, New York, (1997)
- HKK03. Hori, K., Katz, S., Klemm, A., Pandharipande, R., Thomas, R., Vafa, C., Vakil, R., Zaslów, E.: *Mirror symmetry*. Clay Mathematics Monographs 1, American Mathematical Society, Providence, Clay Mathematics Institute, Cambridge, MA, (2003)

- HKP93. Hertz, J., Krough, A., Palmer, R.G.: Introduction to the Theory of Neural Computation. Addison Wesley, (1993)
- HL84. Horsthemke, W., Lefever, R.: Noise-Induced Transitions. Springer, Berlin, (1984)
- HL84. Hamoui, A., Lichnerowicz, A.: Geometry of dynamical systems with time-dependent constraints and time-dependent Hamiltonians: An approach towards quantization. *J. Math. Phys.* **25**, 923, (1984)
- HL86b. Hale, J.K., Lin, X.B.: Examples of transverse homoclinic orbits in delay equations, *Nonlinear Analysis* **10**, 693–709, (1986)
- HL93. Hale, J.K., Lunel, S.M.V.: Introduction to Functional Differential Equations. Springer, New York. (1993)
- HLE01. Hennequin, S., Lefebvre, D., El Moudni, A.: Fuzzy multimodel of timed Petri nets. *IEEE Trans. Syst., Man Cyber. B* **31**, 245–251, (2001)
- HLQ95. Hartley, T.T., Lorenzo, C.F., Qammer, H.K.: Chaos on a Fractional Chua's System, *IEEE Trans. Circ. Sys. Th. App.* **42**, 485–490, (1995)
- HMM95. Hehl, F., McCrea, J., Mielke, E., Ne'eman, Y.: Metric-affine gauge theory of gravity. *Phys. Rep.* **258**, 1, (1995)
- HN54. Huxley, A.F., Niedergerke, R.: Changes in the cross-striations of muscle during contraction and stretch and their structural interpretation. *Nature*, **173**, 973–976, (1954)
- HO01. Hunt, B.R., Ott, E.: Fractal Properties of Robust Strange Nonchaotic Attractors. *Phys. Rev. Lett.* **87**, 254101, (2001)
- HO90. Henkel, R.D., Oppen, M.: Distribution of internal fields and dynamics of neural networks. *Europhys. Lett.* **11**, 403–408, (1990)
- HOY04. Hamaguchi, K., Okada, M., Yamana, M., Aihara, K.: Theory of synfire chain with Mexican-hat connectivity. *IEICE Trans, D-II* **J87**, 1689–1696, (2004)
- HOY05. Hamaguchi, K., Okada, M., Yamana, M., Aihara, K.: Correlated firing in a feed forward network with mexican-hat type connectivity. *Neural Comp.* **17(9)**, 2034–2059, (2005)
- HP96. Hameroff, S.R., Penrose, R.: Orchestrated reduction of quantum coherence in brain microtubules: A model for consciousness. In: Hameroff, S. R., Kaszniak, A.W. and Scott, A.C. Eds: Toward a Science of Consciousness: the First Tucson Discussion and Debates, 507–539. MIT Press, Cambridge, MA, (1996)
- HP96. Hawking, S.W., Penrose, R.: The Nature of Space and Time. Princeton Univ. Press, Princeton, NJ, (1996)
- HP96. Hertz, J., Prügel-Bennett, A.: Learning short synfire chains by self organization. *Network: Computat. Neural Syst.* **7**, 357–363, (1996)
- HS74. Hirsch, M.W., Smale, S.: Differential Equations, Dynamical Systems and Linear Algebra. Academic Press, New York, (1974)

- HS88. Hale, J.K., Sternberg, N.: Onset of Chaos in Differential Delay Equations, *J. Comp. Phys.* **77**(1), 221–239, (1988)
- HS98. Hofbauer, J., Sigmund, K.: Evolutionary games and population dynamics, Cambridge Univ. Press, U.K., (1998)
- HSK92. Hauser, J., Sastry, S., Kokotovic, P.: Nonlinear control via approximate input–output linearization: The ball and beam example, *IEEE Trans. Aut. Con.*, AC–37, 392–398, (1992)
- HT85. Hopfield, J.J., Tank, D.W.: Neural computation of decisions in optimisation problems. *Biol. Cybern.*, **52**, 114–152, (1985)
- HT92. Henneaux, M., Teitelboim, C.: Quantization of Gauge systems. Princeton Univ. Press, (1992)
- HT93. Hunt, L, Turi, J.: A new algorithm for constructing approximate transformations for nonlinear systems. *IEEE Trans. Aut. Con.*, AC–38, 1553–1556, (1993)
- HU79. Hopcroft, J.E., Ullman, J.D.: Introduction to Automata Theory, Languages, and Computation. Addison-Wesley, New York, (1979)
- HV93. Hale, J.K., Verduyn-Lunel, S.M.: Introduction to Functional Differential Equations, *Applied Math. Sci.* **99**, Springer Verlag, New York, (1993)
- HW65. Hubel, D., Wiesel, T.: Receptive fields and functional architecture in two nonstriate visual areas (18 and 19) in the cat. *J. Neurophysiol.* **28**, 229–289, (1965)
- HW83. Van der Heiden, U., Walther, H.-O.: Existence of chaos in control systems with delayed feedback, *J. Diff. Eqs.* **47**, 273–295, (1983)
- HY89. Hassoun, M.H., Youssef, A.M.: A High-Performance Recording Algorithm for Hopfield Model Associative Memories. *Opt. Eng.* **27**(1), 46–54, (1989)
- HYW03. Hao, N., Yildirim, N., Wang, Y., Elston, T.C., Dohlman, H.G.: Regulators of G protein signaling and transient activation of signaling: experimental and computational analysis reveals negative and positive feedback controls on G protein activity. *J. Biol. Chem.*, **278**, 46506–46515, (2003)
- HZ00. Hu, B., Zhou, C.: Phase synchronization in coupled nonidentical excitable systems and array-enhanced coherence resonance. *Phys. Rev. E* **61**, R1001, (2000)
- HZF02. Hahnloser, H., Kozhevnikov, A., Fee, M.: An ultra-sparse code underlies the generation of neural sequences in a songbird. *Nature*, **419**, pp.65–69, 2002. *Nature*, **419**, 65–69, (2002)
- Hag04. Hagras, H.A.: A hierarchical type-2 fuzzy logic control architecture for autonomous mobile robots. *IEEE Trans. Fuz. Sys.* **12**(4), 524–539, (2004)
- Hak00. Haken, H.: Information and Self-Organization: A Macroscopic Approach to Complex Systems. Springer, Berlin, (2000)
- Hak02. Haken, H.: Brain Dynamics, Synchronization and Activity Patterns in Pulse-Coupled Neural Nets with Delays and Noise, Springer, New York, (2002)
- Hak83. Haken, H.: Synergetics: An Introduction (3rd ed). Springer, Berlin, (1983)

- Hak91. Haken, H.: Synergetic Computers and Cognition. Springer-Verlag, Berlin, (1991)
- Hak93. Haken, H.: Advanced Synergetics: Instability Hierarchies of Self-Organizing Systems and Devices (3rd ed). Springer, Berlin, (1993)
- Hak96. Haken, H.: Principles of Brain Functioning: A Synergetic Approach to Brain Activity, Behavior and Cognition, Springer, Berlin, (1996)
- Ham87. Hameroff, S.R.: Ultimate Computing: Biomolecular Consciousness and Nanotechnology. North-Holland, Amsterdam, (1987)
- Har92. Harris-Warrick, R.M. (ed): The Stomatogastric Nervous System. MIT Press, Cambridge, MA, (1992)
- Hat02. Hatcher, A.: Algebraic Topology. Cambridge Univ. Press, Cambridge, (2002)
- Hat77a. Hatze, H.: A myocybernetic control model of skeletal muscle. Biol. Cyber. **25**, 103–119, (1977)
- Hat77b. Hatze, H.: A complete set of control equations for the human musculoskeletal system. J. Biomech. **10**, 799–805, (1977)
- Hat78. Hatze, H.: A general myocybernetic control model of skeletal muscle. Biol. Cyber., **28**, 143–157, (1978)
- Hay01. Haykin, S. (Ed.): Kalman Filtering and Neural Networks. Wiley, New York, (2001)
- Hay91. Haykin, S.: Adaptive Filter Theory. Prentice-Hall, Englewood Cliffs, (1991)
- Hay94. Haykin, S. Neural Networks: A Comprehensive Foundation. Macmillan, (1994)
- Heb49. Hebb, D.O.: The Organization of Behavior, Wiley, New York, (1949)
- Hec77. Heckhausen, H.: Achievement motivation and its constructs: a cognitive model. Motiv. Emot, **1**, 283–329, (1977)
- Hec87. Hecht-Nielsen, R.: Counterpropagation networks. Applied Optics, **26**(23), 4979–4984, (1987)
- Hec90. Hecht-Nielsen, R.: NeuroComputing. Addison-Wesley, Reading, (1990)
- Hee90. Heermann, D.W.: Computer Simulation Methods in Theoretical Physics. (2nd ed), Springer, Berlin, (1990)
- Hen69. Hénon, M.: Numerical study of quadratic area preserving mappings. Q. Appl. Math. **27**, (1969)
- Hen76. Hénon, M.: A two-dimensional mapping with a strange attractor. Com. Math. Phys. **50**, 6977, (1976)
- Her02. Herrero, C.P.: Ising model in small-world networks. Phys. Rev. E **65**, 066110, (2002)
- Hil38. Hill, A.V.: The heat of shortening and the dynamic constants of muscle, Proc. R. Soc. B **76**, 136–195, (1938)
- Hil94. Hilborn, R.C.: Chaos and Nonlinear Dynamics: An Introduction for Scientists and Engineers. Oxford Univ. Press, Oxford, (1994)
- Hir66. Hirzebruch, F.: Topological Methods in Algebraic Geometry, Springer-Verlag, Berlin, (1966)

- Hod64. Hodgkin, A.L.: The Conduction of the Nervous Impulse. Liverpool Univ. Press, Liverpool, (1964)
- Hol75. Holland, J.H.: Adaptation in Natural and Artificial Systems. University of Michigan, Ann Arbor, MI, (1975)
- Hoo86. Hoover, W.G.: Molecular dynamics. Lecture Notes in Physics **258**, Springer, Heidelberg, (1986)
- Hoo95. Hoover, W.G.: Remark on Some Simple Chaotic Flows. Phys. Rev. E, **51**(1), 759–760, (1995)
- Hop82. Hopfield, J.J.: Neural networks and physical systems with emergent collective computational abilities. Proc. Natl. Acad. Sci. USA **79**, 2554, (1982)
- Hop84. Hopfield, J.J.: Neurons with graded response have collective computational properties like those of two-state neurons. Proc. Natl. Acad. Sci. USA, **81**, 3088–3092, (1984)
- Hou79. Houk, J.C.: Regulation of stiffness by skeletomotor reflexes. Ann. Rev. Physiol., **41**, 99–114, (1979)
- Hul00. Hull, J.C.: Options, Futures, and Other Derivatives. (4th ed.), Prentice Hall, New Jersey, (2000)
- Hur93. Hurmuzlu, Y.: Dynamics of bipedal gait. J. Appl. Mech., **60**, 331–343, (1993)
- Hux57. Huxley, A.F.: Muscle structure and theories of contraction. Progr. Biophys. Chem., **7**, 255–328, (1957)
- IAC04. Ikegaya, Y., Aaron, G., Cossart, R., Aronov, D., Lample, I., Ferster, D., Yuste, R.: Synfire chains and cortical songs: temporal modules of cortical activity. Science **304**, 559–564, (2004)
- IAG99. Ivanov, P. Ch., Amaral, L. A. N., Goldberger, A. L., Havlin, S., Rosenblum, M. B., Struzik, Z. & Stanley, H.E.: Multifractality in healthy heartbeat dynamics. Nature, **399**, 461–465, (1999)
- IB05. Ivancevic, V., Beagley, N.: Brain-like functor control machine for general humanoid biodynamics. Int. J. Math. Math. Sci. **11**, 1759–1779, (2005)
- IGF99. Ioffe, L.B., Geshkenbein, V.B., Feigel'man, M.V., Fauchère, A.L., Blatter, G.: Environmentally decoupled sds-wave Josephson junctions for quantum computing, Nature **398**, 679–681, (1999)
- II05. Ivancevic, V., Ivancevic, T.: Human-Like Biomechanics. Springer, Series: Microprocessor-Based and Intelligent Systems Engineering, Vol. 28, (2005)
- II06a. Ivancevic, V., Ivancevic, T.: Natural Biodynamics. World Scientific, Series: Mathematical Biology, (2006)
- II06b. Ivancevic, V., Ivancevic, T.: Geometrical Dynamics of Complex Systems. Springer, Series: Microprocessor-Based and Intelligent Systems Engineering, Vol. 31, (2006)
- IJB99a. Ivancevic, T., Jain, L.C., Bottema, M.: New Two-feature GBAM–Neurodynamical Classifier for Breast Cancer Diagnosis. Proc. KES'99, IEEE Press, (1999)
- IJB99b. Ivancevic, T., Jain, L.C., Bottema, M.: A New Two-Feature FAM–Matrix Classifier for Breast Cancer Diagnosis. Proc. KES'99, IEEE Press, (1999)

- IK04. Ibragimov, N.H., Kolsrud, T.: Lagrangian approach to evolution equations: symmetries and conservation laws, *Nonlinear Dyn.* **36**, 29–40, (2004)
- IK80. Iyanaga, S., Kawada, Y. (eds.): Pontryagin's Maximum Principle. In *Encyclopedic Dictionary of Mathematics*. MIT Press, Cambridge, MA, 295–296, (1980)
- ILI95. Ivancevic, V., Lukman, L., Ivancevic, T.: *Selected Chapters in Human Biomechanics*. Textbook (in Serbian). Univ. Novi Sad Press, Novi Sad, (1995)
- IN92. Igarashi, E., Nogai, T.: Study of lower level adaptive walking in the sagittal plane by a biped locomotion robot. *Advanced Robotics*, **6**, 441–459, (1992)
- IN96. Inoue, M., Nishi, Y.: Dynamical Behavior of Chaos Neural Network of an Associative Schema Model. *Prog. Theoret. Phys.* **95**, 837–850, (1996)
- IP01a. Ivancevic, V., Pearce, C.E.M.: Poisson manifolds in generalised Hamiltonian biomechanics. *Bull. Austral. Math. Soc.* **64**, 515–526, (2001)
- IP01b. Ivancevic, V., Pearce, C.E.M.: Topological duality in humanoid robot dynamics. *ANZIAM J.* **43**, 183–194, (2001)
- IP05b. Ivancevic, V., Pearce, C.E.M.: Hamiltonian dynamics and Morse topology of humanoid robots. *Gl. J. Mat. Math. Sci.* (to appear)
- IS00. Ivancevic, V., Snoswell, M.: *Torson Sports Cybernetics*. Torson Group. Inc., Adelaide, SA, (2000)
- IS01. Ivancevic, V., Snoswell, M.: Fuzzy–stochastic functor machine for general humanoid–robot dynamics. *IEEE Trans. on Sys, Man, Cyber. B*, **31**(3), 319–330, (2001)
- Ich04. Ichinomiya, T.: Frequency synchronization in random oscillator network. *Phys. Rev. E* **70**, 026116, (2004)
- Ich05. Ichinomiya, T.: Path-integral approach to the dynamics in sparse random network. *Phys. Rev. E* **72**, 016109, (2005)
- Ila01. Ilachinski, A.: *Cellular automata*. World Scientific, Singapore, (2001)
- Ing97. Ingber, L.: Statistical mechanics of neocortical interactions: Applications of canonical momenta indicators to electroencephalography, *Phys. Rev. E*, **55**(4), 4578–4593, (1997)
- Ing98. Ingber, L.: Statistical mechanics of neocortical interactions: Training and testing canonical momenta indicators of EEG, *Mathl. Computer Modeling* **27**(3), 33–64, (1998)
- Isi89. Isidori, A.: *Nonlinear Control Systems. An Introduction*, (2nd ed) Springer, Berlin, (1989)
- Isi92. Isichenko, M.B.: Percolation, statistical topography, and transport in random media. *Rev. Mod. Phys.* **64**, 9611043, (1992)
- Ito60. Ito, K.: Wiener Integral and Feynman Integral. *Proc. Fourth Berkeley Symp. Math., Stat., Prob.*, **2**, 227–238, (1960)
- Iva02. Ivancevic, V.: Generalized Hamiltonian biodynamics and topology invariants of humanoid robots. *Int. J. Mat. Mat. Sci.*, **31**(9), 555–565, (2002)

- Iva04. Ivancevic, V.: Symplectic Rotational Geometry in Human Biomechanics. *SIAM Rev.*, **46**(3), 455-474, (2004)
- Iva05a. Ivancevic, V.: Dynamics of Humanoid Robots: Geometrical and Topological Duality. In *Biomathematics: Modeling and simulation* (ed. J.C. Misra), World Scientific, Singapore (to appear)
- Iva05b. Ivancevic, V.: A Lagrangian model in human biomechanics and its Hodge-de Rham cohomology. *Int. J. Appl. Math. Mech.*, **3**, 35-51, (2005)
- Iva05c. Ivancevic, V.: Lie-Lagrangian model for realistic human biodynamics. *Int. J. Humanoid Robotics* (World Sci.) (to appear)
- Iva91. Ivancevic, V.: Introduction to Biomechanical Systems: Modeling, Control and Learning (in Serbian). Scientific Book, Belgrade, (1991)
- Iva95. Ivancevic, T.: Some Possibilities of Multilayered Neural Networks' Application in Biomechanics of Muscular Contractions, Human Motion and Sport Training. Master Thesis (in Serbian), University of Novi Sad, YU, (1995)
- Izh01. Izhikevich, E.M.: Resonate-and-fire neurons. *Neu. Net.*, **14**, 883-894, (2001)
- Izh04. Izhikevich, E.M.: Which model to use for cortical spiking neurons? *IEEE Trans. Neu. Net.*, **15**, 1063-1070, (2004)
- Izh99a. Izhikevich, E.M.: Class 1 neural excitability, conventional synapses, weakly connected networks, and mathematical foundations of pulse-coupled models. *IEEE Trans. Neu. Net.*, **10**, 499-507, (1999)
- Izh99b. Izhikevich, E.M.: Weakly Connected Quasiperiodic Oscillators, FM Interactions, and Multiplexing in the Brain. *SIAM J. Appl. Math.*, **59**(6), 2193-2223, (1999)
- JBC98. Jongen, G., Bollé, D., Coolen, A.C.: The XY spin-glass with slow dynamic couplings. *J. Phys. A* **31**, L737-L742, (1998)
- JBO97. Just, W., Bernard, T., Ostheimer, M., Reibold, E., Benner, H.: Mechanism of time-delayed feedback control. *Phys. Rev. Lett.*, **78**, 203-206, (1997)
- JL99. Juang, C.-F., Lin, C.-T.: A recurrent self-organizing neural fuzzy inference network. *IEEE Trans. Neu. Net.* **10**(4), 828-845, (1999)
- JP98. Jakšić, V., Pillet, C.-A.: Ergodic properties of classical dissipative systems I. *Acta mathematica* **181**, 245-282, (1998)
- JPY96. Jibu, M., Pribram, K.H., Yasue, K.: From conscious experience to memory storage and retrieval: the role of quantum brain dynamics and boson condensation of evanescent photons, *Int. J. Mod. Phys. B*, **10**, 1735, (1996)
- JY95. Jibu, M., Yasue, K.: Quantum brain dynamics and consciousness. John Benjamins, Amsterdam, (1995)
- Jac91. Jackson, E.A.: *Perspectives of Nonlinear Dynamics* **2**. Cambridge Univ. Press, Cambridge, (1991)
- Jos74. Josephson, B.D.: The discovery of tunnelling supercurrents. *Rev. Mod. Phys.* **46**(2), 251-254, (1974)

- Jua02. Juang, C.-F.: A tsf-type recurrent fuzzy network for dynamic systems processing by neural network and genetic algorithms. *IEEE Trans. Fuz. Sys.* **10**(2), 155–170, (2002)
- Jun93. Jung, P.: Periodically driven stochastic systems. *Phys. Reports* **234**, 175, (1993)
- KA01. Kuperman, M., Abramson, G.: Small World Effect in an Epidemiological Model. *Phys. Rev. Lett.* **86**, 2909, (2001)
- KA85. Kawamoto, A.H., Anderson, J.A.: A Neural Network Model of Multistable Perception. *Acta Psychol.* **59**, 35–65, (1985)
- KA98. Kitano, K., Aoyagi, T.: Retrieval dynamics of neural networks for sparsely coded sequential patterns. *J. Phys. A* **31**, L613–L620, (1998)
- KBH91. Kühn, R. S., Bös, S., van Hemmen, J.L.: Statistical mechanics for networks of graded-response neurons. *Phys. Rev. A* **43**, 2084, (1991)
- KD99. Klages, R., Dorfman, J.R.: Simple deterministic dynamical systems with fractal diffusion coefficients. *Phys. Rev. E* **59**, 53615383, (1999)
- KdV95. Korteweg, D.J., de Vries, H.: On the change of form of long waves advancing in a rectangular canal, and on a new type of long stationary waves. *Philosophical Magazine*, **39**, 422–43, (1895)
- KF99. Karlin, I.V., Ferrante, A., Öttinger, H.C.: Perfect Entropy Functions of the Lattice Boltzmann Method. *Europhys. Lett.* **47**, 182–188, (1999)
- KFA69. Kalman, R.E., Falb, P., Arbib, M.A.: *Topics in Mathematical System Theory*. McGraw Hill, New York, (1969)
- KFP91. Kaplan, D.T., Furman, M.I., Pincus, S.M., Ryan, S.M., Lipsitz, L.A., Goldberger, A.L.: Aging and the complexity of cardiovascular dynamics. *Biophys. J.*, **59**, 945–949, (1991)
- KG01. Kumar, R., Garg, V.K.: Control of stochastic discrete event systems modeled by probabilistic languages. *IEEE Trans. Automat. Contr.* **46**, 593–606, (2001)
- KG95. Kumar, R., Garg, V.K.: *Modeling and Control of Logical Discrete Event Systems*. Kluwer, Boston, MA, (1995)
- KH95. Katok, A., Hasselblatt, B.: *Introduction to the Modern Theory of Dynamical Systems*. Cambridge Univ. Press., Cambridge, (1995)
- KHH01. Kim, B.J., Hong, H., Holme, P., Jeong, G.S., Minnhagen, P., Choi, M.Y.: XY models in small-world networks. *Phys. Rev. E* **64**, 056135, (2001)
- KHS93. Koruga, D.L., Hameroff, S.I., Sundareshan, M.K., Withers, J., Loutfy, R.: *Fullerene C60: History, Physics, Nanobiology and Nanotechnology*. Elsevier Science Pub, (1993)
- KK00. Kye, W.-H., Kim, C.-M.: Characteristic relations of type-I intermittency in the presence of noise. *Phys. Rev. E* **62**, 6304–6307, (2000)
- KK02. Kim, E., Kim, S.: Stability analysis and synthesis for an affine fuzzy control system via LMI and ILMI: Continuous case, *IEEE Trans. Fuzzy Syst.* **10**(3), 391400, (2002)

- KK06. Koyama, H., Konishi, T.: Formation of fractal structure in many-body systems with attractive power-law potentials. *Phys. Rev. E* **73**, 016213, (2006)
- KKH03. Kim, J.-W., Kim, S.-Y., Hunt, B., Ott, E.: Fractal properties of robust strange nonchaotic attractors in maps of two or more dimensions. *Phys. Rev. E* **67**, 036211, (2003)
- KL03. Kryzhanovsky, B.V., Litinskii, L.B.: Vector models of associative memory. *Automaton and remote control*, **64**(11), 1782-1793, (2003)
- KL79. Kandel, A., Lee, S.C.: *Fuzzy Switching and Automata: Theory and Applications*. New York: Russak, (1979)
- KLF02. Kryzhanovsky, B.V., Litinskii, L.B., Fonarev, A.: Optical neural network based on the parametrical four-wave mixing process. In: Wang, L., Rajapakse, J.C., Fukushima, K., Lee, S.-Y., and Yao, X. (eds.): *Proc. 9th Int. Conf. Neural Information Processing (ICONIP'02)*, Orchid Country Club, Singapore, **4**, 1704-1707, (2002)
- KLM04. Kryzhanovsky, B.V., Litinskii, L.B., Mikaelian, A.L.: Vector-Neuron Models of Associative Memory. *Neural Networks, Proceedings, IEEE Int. Joint Conf.* (2004)
- KLO03. Kim, S.-Y., Lim, W., Ott, E.: Mechanism for the intermittent route to strange nonchaotic attractors. *Phys. Rev. E* **67**, 056203, (2003)
- KLR03. Kye, W.-H., Lee, D.-S., Rim, S., Kim, C.-M., Park, Y.-J.: Periodic Phase Synchronization in coupled chaotic oscillators. *Phys. Rev. E* **68**, 025201(R), (2003)
- KLV85. Krasil'shchik, I., Lychagin, V., Vinogradov, A.: *Geometry of Jet Spaces and Nonlinear Partial Differential Equations*, Gordon and Breach, Glasgow, (1985)
- KM02. Kryzhanovsky, B.V., Mikaelian, A.L.: On recognition ability of neuronet based on neurons with parametrical frequencies conversion. *Doklady Math.* **65**(2), 286-288, (2002)
- KM89. Kivshar, Y.S., Malomed, B.A.: Dynamics of solitons in nearly integrable systems. *Rev. Mod. Phys.* **61**, 763915, (1989)
- KMK00. Kryzhanovsky, B.V., Mikaelian, A.L., Koshelev, V.N.: On recognition error bound for associative Hopfield memory. *Opt. Mem. Neu. Net.* **9**(4), 267-276, (2000)
- KN00. Kotz, S., Nadarajah, S.: *Extreme Value Distributions*. Imperial College Press, London, (2000)
- KN63/9. Kobayashi, S., Nomizu, K.: *Foundations of Differential Geometry*, Vols. 1,2., Interscience Publ., New York, (1963/1969).
- KO02. Kawamura, M., Okada, M.: Transient dynamics for sequence processing neural networks. *J. Phys. A: Math. Gen.* **35**, 253, (2002)
- KO89. Kamron, N., Olver, P.J.: Le problème d'équivalence à une divergence près dans le calcul des variations des intégrales multiples. *C. R. Acad. Sci. Paris, t.* **308**, Série I, 249-252, (1989)
- KP05. Kobes, R., Peles, S.: A Relationship Between Parametric Resonance and Chaos. *arXiv:nlin/0005005*, (2005)

- KP95. Kocarev, L., Parlitz, U.: General Approach for Chaotic Synchronization with Applications to Communication. *Phys. Rev. Lett.* **74**, 5028–5031, (1995)
- KP96. Kocarev, L., Parlitz, U.: Generalized Synchronization, Predictability, and Equivalence of Unidirectionally Coupled Dynamical Systems. *Phys. Rev. Lett.* **76**, 1816–1819, (1996)
- KR03. Kock, A., Reyes, G.E.: Some calculus with extensive quantities: wave equation. *Theory and Applications of Categories*, **11**(14), 321–336, (2003)
- KR95. Kauffman, L.H., Radford, D.E.: Invariants of 3
- KR99. Kauffman, L.H., Radford, D.E.: Quantum algebra structures on $n \times n$ matrices. *J. Algebra* **213**, 405–436, (1999)
- KS02. Kasabov, N., Song, Q.: Denfis: Dynamic evolving neural fuzzy inference systems and its application for time series prediction. *IEEE Trans. Fuz. Sys.* **10**(2), 144–154, (2002)
- KS03. Kandasamy, V., Smarandache, F.: Fuzzy Cognitive Maps and Neutrosophic Cognitive Maps. Xiquan, Phoenix, (2003)
- KS03. Kavka, C., Schoenauer, M.: Voronoi diagrams based function identification. *Lecture Notes in Computer Science* **2723**, 1089–1100, (2003)
- KS04. Kavka, C., Schoenauer, M.: Evolution of Voronoi based fuzzy controllers. *Lecture Notes in Computer Science* **3242**, 541–550, (2004)
- KS76. Kijowski, J., Szczyrba, W.: A Canonical Structure for Classical Field Theories. *Commun. Math. Phys.* **46**, 183, (1976)
- KS86. Kauffman, S.A. and Smith, R.G.: Adaptive automata based on Darwinian selection. *Physica D*, **22**, 68–82, (1986)
- KS90. Kirchgraber, U., Stoffer, D.: Chaotic behavior in simple dynamical systems, *SIAM Review* **32**(3), 424–452, (1990)
- KS97. Ketoja, J.A., Satija, I.I.: Harper equation, the dissipative standard map and strange non-chaotic attractors: Relationship between an eigenvalue problem and iterated maps. *Physica D* **109**, 70–80, (1997)
- KSJ91. Kandel, E.R., Schwartz, J.H., Jessel, T.M.: Principles of neural sciences. Elsevier, Amsterdam, (1991)
- KSR05. Kavka, C., Schoenauer, M., Roggero, P.: Evolution of Voronoi based Fuzzy Recurrent Controllers. *Proc. of 2005 Conf. on Genetic and Evolut. Comp.* ACM Press, New York, (2005)
- KT01. Kye, W.-H., Topaj, D.: Attractor, bifurcation and on-off intermittency. *Phys. Rev.* **E 63**, 045202(R), (2001)
- KT75. Kamber, F., Tondeur, P.: Foliated Bundles and Characteristic Classes. *Lecture Notes in Math.* 493, Springer, Berlin, (1975)
- KTD03. Kimpo, R., Theunissen, F., Doupe, A.: Propagation of correlated activity through multiple stages of a neural circuit. *J. Neurosci.* **23**, 5750–5761, (2003)
- KV03a. Katic, D., Vukobratovic, M.: Advances in Intelligent Control of Robotic Systems, Book series: Microprocessor-Based and Intelligent Systems Engineering, Kluwer Acad. Pub., Dordrecht, (2003)

- KV03b. Katic, D., Vukobratovic, M.: Survey of intelligent control techniques for humanoid robots. *J. Int. Rob. Sys.*, **37**, 117–141, (2003)
- KV98. Katic, D., Vukobratovic, M.: A neural network-based classification of environment dynamics models for compliant control of manipulation robots. *IEEE Trans. Sys., Man, Cyb., B*, **28**(1), 58–69, (1998)
- KW83. Knobloch, E., Weisenfeld, K.A.: Bifurcations in fluctuating systems: The center manifold approach. *J. Stat. Phys.* **33**, 611, (1983)
- KW97. Krikelis, A., Weems, C.C. (ed): *Associative Processing and Processors*. IEEE Computer Science Press, (1997)
- KY75. Kaplan, J.L., Yorke, J.A.: On the stability of a periodic solution of a differential delay equation, *SIAM J. Math. Ana.* **6**, 268–282, (1975)
- KYO04. Kawamura, M., Yamana, M., Okada, M.: Theory of recurrent neural network with correlated noise. *Tech. Rep. IEICE, NC2003* **103**, 127, (2004)
- KYR98. Kim, C.M., Yim, G.S., Ryu, J.W., Park, Y.J.: Characteristic Relations of Type-III Intermittency in an Electronic Circuit. *Phys. Rev. Lett.* **80**, 5317–5320, (1998)
- KZ91. Kree, R., Zippelius, A.: In *Models of Neural Networks*, I, Domany, R., van Hemmen, J.L., Schulten, K. (Eds), 193, Springer, Berlin, (1991)
- KZ95. Kwon, T.M., Zervakis, M.: KWTa networks and their application. *Multidim. Syst. and Sig. Processing* **6**, 333–346, (1995)
- KZH02. Kiss, I.Z., Zhai, Y., Hudson, J.L.: Emerging coherence in a population of chemical oscillators. *Science* **296**, 1676–1678, (2002)
- Kac51. Kac, M.: On Some Connection between Probability Theory and Differential and Integral Equations. *Proc. 2nd Berkeley Sympos. Math. Stat. and Prob.*, 189–215, (1951)
- Kai99. Kaiser, D.: Larger domains from resonant decay of disoriented chiral condensates. *Phys. Rev. D* **59**, 117901, (1999)
- Kal60. Kalman, R.E.: A new approach to linear filtering and prediction problems. *Transactions of the ASME, Ser. D, J. Bas. Eng.*, **82**, 34–45, (1960)
- Kam92. Van Kampen, N.G.: *Stochastic Processes in Physics and Chemistry*. North-Holland, Amsterdam, (1992)
- Kan58. Kan, D.M.: Adjoint Functors. *Trans. Am. Math. Soc.* **89**, 294–329, (1958)
- Kan88. Kanter, I.: Potts-glass models of neural networks. *Phys. Rev. A* **37**, 2739–2742, (1988)
- Kan98. Kanatchikov, I.: Canonical structure of classical field theory in the polymomentum phase space. *Rep. Math. Physics*, **41**(1), 49–90, (1998)
- Kap00. Kaplan, D.T.: Applying Blind Chaos Control to Find Periodic Orbits. *arXiv:nlin.CD/0001058*, (2000)
- Kap05. Kappen, H.J.: A linear theory for control of nonlinear stochastic systems. *Phys. Rev. Lett.* (to appear)

- Kar00. M. Karplus: Aspects of Protein Reaction Dynamics: Deviations from Simple Behavior. *J. Phys. Chem. B* **104**, 11-27, (2000)
- Kas02. Kasabov, N.: *Evolving connectionist systems: Methods and applications in bioinformatics, brain study and intelligent machines*. Springer, London, (2002)
- Kau94. Kauffman, L.H.: *Knots and Physics*, (2nd ed.) World Scientific, Singapore (1994)
- Kha92. Khalil, H.K.: *Nonlinear Systems*. New York: MacMillan, (1992)
- Khi57. Khinchin, A.I.: *Mathematical foundations of Information theory*. Dover, (1957)
- Kla00. Klauder, J.R.: *Beyond Conventional Quantization*, Cambridge Univ. Press, Cambridge, (2000)
- Kla97. Klauder, J.R.: Understanding Quantization. *Found. Phys.* **27**, 1467–1483, (1997)
- Koc81. Kock, A.: *Synthetic Differential Geometry*, London Math.Soc. Lecture Notes Series No. 51, Cambridge Univ. Press, Cambridge, (1981)
- Kog79. Kogut, J.: An introduction to lattice gauge theory and spin systems. *Rev. Mod. Phys.* **51**, 659-713, (1979)
- Koh82. Kohonen, T.: Self-Organized Formation of Topologically Correct Feature Maps. *Biological Cybernetics* **43**, 59–69, (1982)
- Koh88. Kohonen, T.: *Self Organization and Associative Memory*. Springer, (1988)
- Koh91. Kohonen, T.: *Self-Organizing Maps: Optimization Approaches*. In: *Artificial Neural Networks*, ed. T. Kohonen et al. North-Holland, Amsterdam, (1991)
- Kos04. Kosmann-Schwarzbach, Y.: Derived brackets. *Lett. Math. Phys.* **69**, 61–87, (2004)
- Kos86. Kosko, B.: Fuzzy Cognitive Maps. *Int. J. Man-Mach. Stud.* **24**, 65-75, (1986)
- Kos88. Kosko, B.: Bidirectional Associative Memory. *IEEE Trans. Sys. Man Cyb.* **18**, 49–60, (1988)
- Kos92. Kosko, B.: *Neural Networks and Fuzzy Systems, A Dynamical Systems Approach to Machine Intelligence*. Prentice-Hall, New York, (1992)
- Kos93. Kosko, B.: *Fuzzy Thinking*. Disney Books, Hyperion, (1993)
- Kos96. Kosko, B.: *Fuzzy Engineering*. Prentice Hall, New York, (1996)
- Kos99. Kosko, B.: *The Fuzzy Future: From Society and Science to Heaven in a Chip*. Random House, Harmony, (1999)
- Koz92. Koza, J.R.: *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, (1992)
- Kre84. Krener, A.: Approximate linearization by state feedback and coordinate change, *Systems Control Lett.*, 5, 181–185, (1984)
- Kru97. Krupkova, O.: *The Geometry of Ordinary Variational Equations*. Springer, Berlin, (1997)

- Kub57. Kubo, R.: Statistical-mechanical theory of irreversible processes. I. *J. Phys. Soc. (Japan)* **12**, 570-586, (1957)
- Kuh85. Kuhl, J.: Volitional Mediator of cognition-Behaviour consistency: Self-regulatory Processes and action versus state orientation (pp. 101-122). In: J. Kuhl & S. Beckman (Eds.) *Action control: From Cognition to Behaviour*. Springer, Berlin, (1985)
- Kur76. Kuramoto, Y., Tsuzuki, T.: Persistent propagation of concentration waves in dissipative media far from thermal equilibrium. *Progr. Theor. Physics* **55**, 365, (1976)
- Kur84. Kuramoto, Y.: *Chemical Oscillations. Waves and Turbulence*. Springer, New York, (1984)
- LAT83. Lobb, C.J., Abraham, D.W., Tinkham, M.: Theoretical interpretation of resistive transitions in arrays of superconducting weak links. *Phys. Rev.* **B27**, 150, (1983)
- LC90. Lafortune, S., Chen, E.: The infimal closed controllable superlanguage and its application in supervisory control. *IEEE Trans. Automat. Contr.* **35**, 398-405, (1990)
- LCD87. Leggett, A.J., Chakravarty, S., Dorsey, A.T., Fisher, M.P.A., Chang, A., Zwerger, W.: Dynamics of the dissipative two-state system. *Rev. Mod. Phys.* **59**, 1, (1987)
- LD05. Levine, M., Davidson, E.H.: Gene regulatory networks for development. *Proc Natl. Acad. Sci. USA*, **102**(14), 4936-4942, (2005)
- LD98. Loss, D., DiVincenzo, D.P.: Quantum computation with quantum dots. *Physical Review A* **57**(1), 120-126, (1998)
- LGS98. Lygeros, J., Godbole, D.N., Sastry, S.: Verified hybrid controllers for automated vehicles, *IEEE Trans. Aut. Con.*, **43**, 522-539, (1998)
- LL59. Landau, L.D., Lifshitz, E.M.: *Fluid Mechanics*. Pergamon Press, (1959)
- LL78. Landau, L.D., Lifshitz, E.M.: *Statistical Physics*. Pergamon Press, Oxford, (1978)
- LL86. Lin, C.T., Lee, S.G.: *Neural Fuzzy Systems: A Neural-Fuzzy Synergism to Intelligent Systems*. Prentice Hall, Englewood Cliffs, NJ, (1986)
- LL98. Labastida, J.M.F., Lozano, C.: Lectures on topological quantum field theory, in *Proceedings of the CERN-Santiago de Compostela-La Plata Meeting on 'Trends in Theoretical Physics'*, eds. H. Falomir, R. Gamboa, F. Schaposnik, Amer. Inst. Physics, New York, (1998)
- LLY02. Liu, Z.H., Lai, Y-C., Ye, N., Dasgupta, P.: Connectivity distribution and attack tolerance of general networks with both preferential and random attachments. *Phys. Lett. A* **303**, 337344, (2002)
- LM97. Lewis, A.D., Murray, R.M.: Controllability of simple mechanical control systems, *SIAM J. Con. Opt.*, **35**(3), 766-790, (1997)
- LM99. Lewis, A.D., Murray, R.M.: Configuration controllability of simple mechanical control systems, *SIAM Review*, **41**(3), 555-574, (1999)

- LO97. Lenz, H., Obradovic, D.: Robust control of the chaotic Lorenz system. *Int. J. Bif. Chaos* **7**, 2847–2854, (1997)
- LP94. Langer, J., Perline, R.: Local geometric invariants of integrable evolution equations, *J. Math. Phys.*, **35**(4), 1732–1737, (1994)
- LRR02. Lee, T.I., Rinaldi, N.J., Robert, F., Odom D.T., Bar-Joseph Z., et al.: Transcriptional regulatory networks in *Saccharomyces cerevisiae*. *Science*, **298**, 799–804, (2002)
- LS97. Louko, J., Sorkin, R.D.: Complex actions in two-dimensional topology change. *Class. Quant. Grav.* **14**, 179–204, (1997)
- LSA03. Louzoun, Y., Solomon, S., Atlan, H., Cohen, I.R.: Proliferation and competition in discrete biological systems. *Bull. Math. Biol.* **65**, 375, (2003)
- LT00. Lee, C.-H., Teng, C.-C.: Identification and control of dynamic systems using recurrent fuzzy neural networks. *IEEE Trans. Fuz. Sys.* **8**(4), 349–366, (2000)
- LV97. Li, M., Vitanyi, P.: *An Introduction to Kolmogorov Complexity and its Applications*. Springer, New York, (1997)
- LVW89. Lerche, W., Vafa, C., Warner, N.P.: Chiral Rings In N=2 Superconformal Theories, *Nucl. Phys. B* **324**, 427, (1989)
- LW05. Lipan, O., Wong, W.H.: The use of oscillatory signals in the study of genetic networks. *Proc. Natl. Acad. Sci. USA*, **102**, 7063–7068, (2005)
- LW93. Lawford, M., Wonham, W.M.: Supervisory control of probabilistic discrete event systems. In *Proc. 36th Midwest Symp. Circ. Sys.*, 327–331, (1993)
- LW95. Lani-Wayda, B, Walther, H.-O.: Chaotic motion generated by delayed negative feedback, Part I: A transversality criterion. *Diff. Int. Eqs.* **8**(6), 1407–52, (1995)
- LW96. Lani-Wayda, B, Walther, H.-O.: Chaotic motion generated by delayed negative feedback, Part II: Construction of nonlinearities. *Math. Nachr.* **180**, 141–211, (1996)
- LY01. Lin, F., Ying, H.: Fuzzy discrete event systems and their observability, in *Proc. Joint Int. Conf. 9th Int. Fuzzy Systems Assoc. World Congr. 20th North Amer. Fuzzy Inform. Process. Soci.*, 1271–1276, Vancouver, BC, Canada, (2001)
- LY02. Lin, F., Ying, H.: Modeling and control of fuzzy discrete event systems. *IEEE Trans. Syst., Man, Cybern.*, **B 32**, 408–415, (2002)
- LY77. Li, T.Y., Yorke, J.A.: Period three implies chaos. *Am. Math. Monthly* **82**, 985–992, (1977)
- LY85. Ledrappier, F., Young, L.S.: The metric entropy of diffeomorphisms: I. Characterization of measures satisfying Pesin’s formula, II. Relations between entropy, exponents and dimension. *Ann. of Math.* **122**, 509–539, 540–574, (1985)
- Lai94. Lai, Y.-C.: Controlling chaos. *Comput. Phys.*, **8**, 62–67, (1994)
- Lak03. Lakshmanan, M., Rajasekar, S: *Nonlinear Dynamics: Integrability, Chaos and Patterns*, Springer-Verlag, New York, (2003)

- Lak97. Lakshmanan, M.: Bifurcations, Chaos, Controlling and Synchronization of Certain Nonlinear Oscillators. In *Lecture Notes in Physics*, **495**, 206, Y. Kosmann-Schwarzbach, B. Grammaticos, K.M. Tamizhmani (ed), Springer-Verlag, Berlin, (1997)
- Lam76. Lamb, G.L. Jr.: Bäcklund transforms at the turn of the century. In *Bäcklund Transforms*, (ed) by R.M. Miura, Springer, Berlin, (1976)
- Lam94. Lamport, L.: The temporal logic of actions. *ACM Transactions on Programming Languages and Systems*, **16**(3), 872–923, (1994)
- Lan00. Landsberg, A.S.: Disorder-induced desynchronization in a 2×2 circular Josephson junction array. *Phys. Rev. B* **61**, 36413648, (2000)
- Lan95. Landsman, N.P.: Against the Wheeler–DeWitt equation. *Class. Quan. Grav. L* **12**, 119–124, (1995)
- Lan95. Lani–Wayda, B: Hyperbolic Sets, Shadowing and Persistence for Noninvertible Mappings in Banach spaces, *Research Notes in Mathematics* **334**, Longman Group Ltd., Harlow, Essex, (1995)
- Las77. Lasota, A.: Ergodic problems in biology. *Asterisque* **50**, 239–250, (1977)
- Lav03. Lavrov, A.: Fuzzy extension of discrete event supervisory control. In *Proc. 2003 IEEE Int. Symp. Comp. Intell. Robot. Automat.*, 1138–1143, Kobe, Japan, (2003)
- Leb93. Lebowitz, J.L.: Boltzmann’s entropy and time’s arrow. *Physics Today* **46**(9), 32–38, (1993)
- Led81. Ledrappier, F.: Some Relations Between Dimension and Lyapunov Exponents. *Commun. Math. Phys.* **81**, 229–238, (1981)
- Lee90. Lee, C.C.: Fuzzy Logic in Control Systems. *IEEE Trans. Sys., Man, Cybern.*, **20**(2), 404–435, (1990)
- Lei02. Leinster, T.: A survey of definitions of n –category. *Theor. Appl. Categ.* **10**, 1–70, (2002)
- Lei03. Leinster, T.: *Higher Operads, Higher Categories*, London Mathematical Society Lecture Notes Series, Cambridge Univ. Press, (2003)
- Lei04. Leinster, T.: Operads in higher-dimensional category theory. *Theor. Appl. Categ.* **12**, 73–194, (2004)
- Lew00a. Lewis, A.D.: Simple mechanical control systems with constraints, *IEEE Trans. Aut. Con.*, **45**(8), 1420–1436, (2000)
- Lew00b. Lewis, A.D.: Affine connection control systems. *Proceedings of the IFAC Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control* 128–133, Princeton, (2000)
- Lew51. Lewin, K.: *Field Theory in Social Science*. Univ. Chicago Press, Chicago, (1951)
- Lew95. Lewis, A.D.: Aspects of Geometric Mechanics and Control of Mechanical Systems. Technical Report CIT-CDS 95-017 for the Control and Dynamical Systems Option, California Institute of Technology, Pasadena, CA, (1995)

- Lew97. Lewin, K.: Resolving Social Conflicts: Field Theory in Social Science, American Psych. Assoc., New York, (1997)
- Lew99. Lewis, A.D.: When is a mechanical control system kinematic?, in Proceedings of the 38th IEEE Conf. Decis. Con., 1162–1167, IEEE, Phoenix, AZ, (1999)
- Li04a. Li, Y.: Chaos in Partial Differential Equations. Int. Press, Somerville, MA, (2004)
- Li04b. Li, Y.: Homoclinic tubes and chaos in perturbed sine-Gordon equation. Cha. Sol. Fra., **20**(4), 791–798, (2004)
- Lia83. Liao, S.T.: On hyperbolicity properties of nonwandering sets of certain 3-dimensional differential systems. Acta Math. Sc., **3**, 361–368, (1983)
- Lig58. Lighthill, M.J.: An Introduction to Fourier Analysis and Generalised Functions. Cambridge Univ. Press, (1958)
- Lin89. Linsker, R.: How To generate Ordered maps by Maximizing the Mutual Information between Input and Output Signals. Neural Computation **1**, 402–411, (1989)
- Lin97. Linz, S.J.: Nonlinear Dynamical Models and Jerky Motion. Am. J. Phys., **65**(6), 523–526, (1997)
- Lol01. Loll, R.: Discrete Lorentzian quantum gravity. Nucl. Phys. B **94**, 96–107, (2001)
- Lol98. Loll, R.: Discrete approaches to quantum gravity in four dimensions. Living Reviews in Relativity, **13**, (1998)
- Loo00. Loo, K.: A Rigorous Real Time Feynman Path Integral and Propagator. J. Phys. A: Math. Gen, **33**, 9215–9239, (2000)
- Loo99. Loo, K.: A Rigorous Real Time Feynman Path Integral. J. Math. Phys., **40**(1), 64–70, (1999)
- Lor63. Lorenz, E.N.: Deterministic Nonperiodic Flow. J. Atmos. Sci., **20**, 130–141, (1963)
- Lou83. Louie, A.H.: (a) Categorical system theory and the phenomenological calculus. Bull. Math. Biol., **45**, 1029–1045; (b), Categorical system theory. Bull. Math. Biol., **45**, 1047–1072, (1983)
- Lug02. Luger, G.F.: Artificial Intelligence: Structures and Strategies for Complex Problem Solving. Pearson Educ (4th ed.). Ltd, Harlow, UK, (2002)
- Lui02. Luinge, H.J.: Inertial Sensing of Human Movement. Ph.D. thesis Univ. Twente, Twente Univ. Press, (2002)
- Lya56. Lyapunov, A.M.: A general problem of stability of motion. Acad. Sc. USSR, (1956)
- MA02. Mascaldi, M. *et al.*: Proton MR Spectroscopy of the Cerebellum and Pons in Patients with Degenerative Ataxia, Radiology, **223**, 371, (2002)
- MAK04. Morelli, L.G., Abramson, G., Kuperman, M.N.: Associative memory on a small-world neural network. Eur. Phys. J. B **38**, 495–500, (2004)
- MBP00. Myung, I.J., Balasubramanian, V., Pitt, M.A.: Counting probability distributions: differential geometry and model selection. Proceedings of the National Academy of Science, USA, **97**, 11170–11175, (2000)

- MDC85. Martinis, J.M., Devoret, M.H., Clarke, J.: Energy-Level Quantization in the Zero-Voltage State of a Current-Biased Josephson Junction. *Phys. Rev. Lett.* **55**, 1543-1546, (1985)
- MDT00. Mangioni, S.E., Deza, R.R., Toral, R., Wio, H.S.: Non-equilibrium phase transitions induced by multiplicative noise: effects of self-correlation. *Phys. Rev. E*, **61**, 223-231, (2000)
- MF00. Mel, B., Fiser, J.: Minimizing Binding Errors Using Learned Conjunctive Features. *Neural Comp.* **12**, 731-762, (2000)
- MFB00. Myung, I.J., Forster, M., Browne, M.W.: Special issue on model selection. *J. Math. Psych.*, **44**, 1-2, (2000)
- MFV90. Morandi, G., Ferrario, C., Lo Vecchio, G., Marmo, G., Rubano, C.: The inverse problem in the calculus of variations and the geometry of the tangent bundle, *Phys. Rep.* **188**, 147, (1990)
- MG02. Martínez, H.B., Skarmeta, A.G.: A framework for defining and learning fuzzy behaviors for autonomous mobile robots. *Int. J. Intell. Syst.* **17**, 1-20, (2002)
- MG77. Mackey, M.C., Glass, L.: Oscillation and chaos in physiological control systems. *Science* **197**, 287-295, (1977)
- MGB96. Mira, C., Gardini, L., Barugola, A., Cathala, J.-C.: *Chaotic Dynamics in Two-Dimensional Noninvertible Maps*. World Scientific, Singapore, (1996)
- MH38. Morse, M., Hedlund, G.: *Symbolic Dynamics*. *Am. J. Math.* **60**, 815-866, (1938)
- MK05. Moon, S.J., Kevrekidis, I.G.: An equation-free approach to coupled oscillator dynamics: the Kuramoto model example. Submitted to *Int. J. Bifur. Chaos*, (2005)
- MKL. Mikaelian, A.L., Kryzhanovsky, B.V., Litinskii, L.B.: Parametrical Neural Network. *Opt. Mem. Neu. Net.* **12**(3), 227-236, (2003)
- MKO03. Mimura, K., Kimoto, T., Okada, M.: Synapse efficiency diverges due to synaptic pruning following over-growth. *Phys. Rev. E* **68**, 031910, (2003)
- MKO04. Mimura, K., Kawamura, M., Okada, M.: The path-integral analysis of an associative memory model storing an infinite number of finite limit cycles. *J. Phys. A: Math. Gen.* **37**, 6437-6454, (2004)
- ML81. Morris, C., Lecar, H.: Voltage oscillations in the barnacle giant muscle fiber. *Biophys. J.*, **35**, 193-213, (1981)
- MLS94. Murray, R.M., Li, X., Sastry, S.: *Robotic Manipulation*, CRC Press, Boca Raton, FL, (1994)
- MM75. Marsden, J.E., McCracken, M.: *The Hopf bifurcation and its applications*. MIT Press, Cambridge, MA, (1975)
- MM92. Marathe, K., Martucci, G.: *The Mathematical Foundations of Gauge Theories*. North-Holland, Amsterdam, (1992)
- MM95. Matsui, N., Mori, T.: The efficiency of the chaotic visual behavior in modeling the human perception-alternation by artificial neural network. In *Proc. IEEE ICNN'95*. **4**, 1991-1994, (1995)

- MM97. Mouzouris, G., Mendel, J.: Dynamic non-singleton fuzzy logic systems for nonlinear modeling. *IEEE Trans. Fuz. Sys.* **5**(2), 199–208, (1997)
- MN95a. Mavromatos, N.E., Nanopoulos, D.V.: A Non-critical String (Liouville) Approach to Brain Microtubules: State Vector reduction, Memory coding and Capacity. ACT-19/95, CTP-TAMU-55/95, OOTP-95-52P, (1995)
- MN95b. Mavromatos, N.E., Nanopoulos, D.V.: Non-Critical String Theory Formulation of Microtubule Dynamics and Quantum Aspects of Brain Function. ENSLAPP-A-524/95, (1995)
- MNL03. Motter, A.E., Nishikawa, T., Lai, Y-C.: Large-scale structural organization of social networks. *Phys. Rev. E* **68**, 036105, (2003)
- MNM02. Montagna, G., Nicosini, O., Moreni, N.: A path integral way to option pricing. *Physica A* **310**, 450–466, (2002)
- MO95. Mimura, K., Okada, M.: The S/N analysis of associative memory: sign alternating and paring memorization method. *JNNS Tran.* **2**, 38, (1995)
- MOK96. Mimura, K., Okada, M., Kurata, K.: Capacity of autocorrelation associative memory with quantized synaptic weight. *Sys. Comp. Jpn.* **27**, 55, (1996)
- MOK98a. Mimura, K., Okada, M., Kurata, K.: Robustness to noise of associative memory using nonmonotonic analogue neurons. *IEICE Trans. Inf. Sys.* **E81-D**, 928, (1998)
- MOK98b. Mimura, K., Okada, M., Kurata, K.: Associative memory model with forgetting process using nonmonotonic neurons. *IEICE Trans. Inf. Sys.* **E81-D**, 1298, (1998)
- MP01. Morales, C.A., Pacifico, M.J.: Mixing attractors for 3-flows. *Nonlinearity*, **14**, 359–378, (2001)
- MP03. Myung, I.J., Pitt, M.A.: Model Evaluation, Testing and Selection. To appear in K. Lambert and R. Goldstone (eds.), *Handbook of Cognition*, Sage Publ., (2003)
- MP43. McCulloch W., Pitts W.: A logical calculus of the ideas imminent in the nervous activity. *Bull. Math. Biophys.* **5**, 115–133, 1943
- MP69. Minsky, M., Papert, S.: *Perceptrons*. MIT Press, Cambridge, MA, (1969)
- MP92. De Melo, W., Palis, J.: *Geometric Theory of Dynamical Systems-An Introduction*. Springer Verlag, Berlin, (1982)
- MPR98. McLeod, P., Plunkett, K., Rolls, E.T.: *Introduction to connectionist modeling of cognitive processes*. Oxford Univ. Press, Oxford, (1998)
- MPT98. Milanović, Lj., Posch, H.A., Thirring, W.: Statistical mechanics and computer simulation of systems with attractive positive power-law potentials. *Phys. Rev. E* **57**, 27632775, (1998)
- MPV87. Mézard, M., Parisi, G., Virasoro, M.A.: *Spin Glass Theory and Beyond*. World Scientific, Singapore, (1987)
- MR99. Marsden, J.E., Ratiu, T.S.: *Introduction to Mechanics and Symmetry: A Basic Exposition of Classical Mechanical Systems*. (2nd ed), Springer, New York, (1999)

- MRS04. Munteanu, F., Rey, A.M., Salgado, M.: The Günther's formalism in classical field theory: momentum map and reduction. *J. Math. Phys.* **45**(5), 1730–1750, (2004)
- MRS95. Müller, B., Reinhardt, J., Strickland, M.T.: *Neural Networks, An Introduction*. Springer, (1995)
- MS03. Moroianu, A., Semmelmann, U.: Twistor forms on Kähler manifolds. *Ann. Scuola Norm. Sup. Pisa Cl. Sci.* **2**(4), 823–845, (2003)
- MS78. Modugno, M., Stefani, G.: Some results on second tangent and cotangent spaces. *Quadern dell' Istituto di Matematica dell' Universit a di Lecce Q.*, **16**, (1978)
- M-PS90. Mallet–Paret, J., Smith, H.L.: The Poincaré–Bendixson theorem for monotone cyclic feedback systems, *J. Dyn. Diff. Eqns.* **2**, 367–421, (1990)
- MS90. Mirollo, R.E., Strogatz, S.H.: Synchronization of pulse-coupled biological oscillators. *SIAM J. Appl. Math.* **50**, 1645, (1990)
- MS95. Marmo, G., Simoni, A., Stern, A.: Poisson Lie group symmetries for the isotropic rotator. *Int. J. Mod. Phys. A* **10**, 99–114, (1995)
- MS96. Mallet–Paret, J., Sell, G.: The Poincaré–Bendixson theorem for monotone cyclic feedback systems with delay, *J. Diff. Eqns.* **125**, 441–489, (1996)
- MT02. Mastorocostas, P.A., Theocharis, J.B.: A recurrent fuzzy-neural model for dynamic system identification. *IEEE Trans. Sys. Man Cyber.* **32**(2), 176–190, (2002)
- MTW73. Misner, C.W., Thorne, K.S., Wheeler, J.A.: *Gravitation*. Freeman, San Francisco, (1973)
- MW89. Marcus, C.M., Westervelt, R.M.: Dynamics of iterated-map neural networks. *Phys. Rev. A* **40**(1), 501–504, (1989)
- MW94. Mallet–Paret, J., Walther, H.-O.: Rapid oscillations are rare in scalar systems governed by monotone negative feedback with a time lag. *Math. Inst. Univ. Giessen*, (1994)
- MWH01. Michel, A.N., Wang, K., Hu, B.: *Qualitative Theory of Dynamical Systems* (2nd ed). Dekker, New York, (2001)
- MZA03. Mangan, S., Zaslaver, A., Alon, U.: The coherent feedforward loop serves as a sign-sensitive delay element in transcription networks. *JMB*, **334**(2), 197–204, (2003)
- MZK05. Motter, A.E., Zhou, C., Kurths, J.: Enhancing complex-network synchronization. *Europhys. Lett.*, **69**(3), 334–340, (2005)
- MacL71. MacLane, S.: *Categories for the Working Mathematician*. Springer, New York, (1971)
- Mal798. Malthus, T.R.: *An essay on the Principle of Population*. (Originally published in 1798.) Penguin, (1970)
- Mal88. Mallet–Paret, J.: Morse decompositions for delay differential equations. *J. Diff. Eqns.* **72**, 270–315, (1988)
- Mal00. Malasoma, J.M.: What is the Simplest Dissipative Chaotic Jerk Equation which is Parity Invariant? *Phys. Lett. A*, **264**(5), 383–389, (2000)

- Man80a. Mandelbrot, B.: Fractal aspects of the iteration of $z \mapsto \lambda z(1 - z)$ for complex λ, z , *Annals NY Acad. Sci.* **357**, 249–259, (1980)
- Man80b. Mandelbrot, B.: *The Fractal Geometry of Nature*. WH Freeman and Co., New York, (1980)
- Man82. Mañé, R.: An ergodic closing lemma. *Annals of Math.*, **116**, 503–540, (1982)
- Man98. Manikonda, V.: *Control and Stabilization of a Class of Nonlinear Systems with Symmetry*. PhD Thesis, Center for Dynamics and Control of Smart Structures, Harvard Univ., Cambridge, MA, (1998)
- Mar98. Marieb, E.N.: *Human Anatomy and Physiology*. (4th ed.), Benjamin/Cummings, Menlo Park, CA, (1998)
- Mar99. Marsden, J.E.: *Elementary Theory of Dynamical Systems*. Lecture notes. CDS, Caltech, (1999)
- Mat01. Matthews, G.G.: *Neurobiology: Molecules, Cells and Systems* (2nd ed), Blackwell Sci. Inc., (2001)
- Mat02. Matacz, A.: Path Dependent Option Pricing: the Path Integral Partial Averaging Method. *J. Comp. Finance*, **6**(2), (2002)
- May73. May, R.M. (ed.): *Stability and Complexity in Model Ecosystems*. Princeton Univ. Press, Princeton, NJ, (1973)
- May76a. May, R.M. (ed.): *Theoretical Ecology: Principles and Applications*. Blackwell Sci. Publ. (1976)
- May76b. May, R.: Simple Mathematical Models with Very Complicated Dynamics. *Nature*, **261**(5560), 459–467, (1976)
- May79. Maybeck, P.S.: *Stochastic Models, Estimation, and Control*, Volume 1, Academic Press, New York, (1979)
- May81. Mayer, P.A.: *A Differential Geometric Formalism for the Ito Calculus*. Lecture Notes in Mathematics, Vol. 851, Springer, New York, (1981)
- McCul87. McCullagh, P.: *Tensor methods in statistics. Monographs in statistics and applied probability*. Chapman & Hall, Cambridge, UK, (1987)
- Men98. Mendes, R.V.: Conditional exponents, entropies and a measure of dynamical self-organization. *Phys. Let. A* **248**, 167–173, (1998)
- Mer73. Merton, R.: Theory of Rational Option Pricing. *Bell J. Econom. Managem. Sci.* **4**, 141–183, (1973)
- Met97. Metzger, M.A.: Applications of nonlinear dynamical systems theory in developmental psychology: Motor and cognitive development. *Nonlinear Dynamics, Psychology, and Life Sciences*, **1**, 55–68, (1997)
- Mey70. Meyer, K.R.: Generic bifurcation of periodic points. *Trans. Amer. Math. Soc.* **149**, 95–107, (1970)
- Mey71. Meyer, K.R.: Generic stability properties of periodic points, *Trans. Amer. Math. Soc.* **154**, 273–277, (1971)
- Mey73. Meyer, K.R.: Symmetries and integrals in mechanics. In *Dynamical Systems* (M. Peixoto, Ed.), 259–272, Academic Press, New York, (1973)

- Mic01. Michor, P.W.: Topics in Differential Geometry. Lecture notes of a course in Vienna, (2001)
- Mic04. Michel, O.: Kephra simulator package version 2.0: Freeware mobile robot simulator. <http://wwwi3s.unice.fr/om/khep-sim.html>, (2004)
- Mil56. Miller, G.A.: The magical number seven, plus or minus two: Some limits on our capacity for processing information, *Psych. Rev.*, **63**, 81–97, (1956)
- Mil63. Milnor, J.: Morse Theory. Princeton Univ. Press, Princeton, (1963)
- Mil65. Milnor, J.: Lectures on the H–Cobordism Theorem. Math. Notes. Princeton Univ. Press, Princeton, (1965)
- Mit96. Mitchell, M.: An Introduction to Genetic Algorithms. MIT Press, Cambridge, MA, (1996)
- Mla91. Mladenova, C.: Mathematical Modeling and Control of Manipulator Systems. *Int. J. Rob. Comp. Int. Man.*, **8**(4), 233–242, (1991)
- Mla99. Mladenova, C.: Applications of Lie Group Theory to the Modeling and Control of Multibody Systems. *Mult. Sys. Dyn.*, **3**(4), 367–380, (1999)
- Moh91. Mohler, R.R.: Nonlinear systems, Vol. 2. Applications to bilinear control. Prentice Hall, Inc. (1991)
- Mor34. Morse, M.: The Calculus of Variations in the Large. Amer. Math. Soc. Coll. Publ. No. 18, Providence, RI, (1934)
- Mos96. Mosekilde, E.: Topics in Nonlinear Dynamics: Application to Physics, Biology and Economics. World Scientific, Singapore, (1996)
- Mur02. Murray, J.D.: Mathematical Biology, Vol. I: An Introduction (3rd ed.), Springer, New York, (2002)
- Mus99. Mustafa, M.T.: Restrictions on harmonic morphisms. *Conformal Geometry and Dynamics* (AMS), **3**, 102–115, (1999)
- NAY60. Nagumo, J., Arimoto, S., Yoshizawa, S.: An active pulse transmission line simulating 1214-nerve axons, *Proc. IRL* **50**, 2061–2070, (1960)
- NE99. Nimmo, S., Evans, A.K.: The Effects of Continuously Varying the Fractional Differential Order of Chaotic Nonlinear Systems. *Chaos, Solitons & Fractals* **10**, 1111–1118, (1999)
- NF00. Nolfi, S., Floreano, D.: Evolutionary Robotics, The Biology, Intelligence, and Technology of Self-Organizing Machines. Bradford Books, (2000)
- NF91. Nitta, T., Furuya, T.: A complex back-propagation learning. *Trans. Inf. Proc. Soc. Jpn.*, **32**(10), 1319–1329, (1991)
- NHP89. Noakes, L., Heinzinger, G., Paden, B.: Cubic splines on curved spaces, *IMA J. Math. Con. Inf.*, **6**(4), 465–473, (1989)
- NK96. Nishikawa, T., Kaneko, K.: Fractalization of a torus as a strange nonchaotic attractor. *Phys. Rev. E* **54**, 61146124, (1996)
- NKF97. Nishimura, H., Katada, N., Fujita, Y.: Dynamic Learning and Retrieving Scheme Based on Chaotic Neuron Model, In:

- R. Nakamura et al. (eds): Complexity and Diversity, Springer-Verlag, New York, 64–66, (1997)
- NLG00. Newrock, R.S., Lobb, C.J., Geigenmüller, U., Octavio, M.: Solid State Physics. Academic Press, San Diego, Vol. 54, (2000)
- NM44. von Neumann, J., Morgenstern, O.: Theory of Games and Economic Behavior. Princeton Univ. Press, Princeton, NJ, (1944)
- NML02. Nishikawa, T., Motter, A.E., Lai, Y.-C., Hoppensteadt, F.C.: Smallest small-world network. Phys. Rev. E **66**, 046139, (2002)
- NML03. Nishikawa, T., Motter, A.E., Lai, Y.C., Hoppensteadt, F.C.: Heterogeneity in Oscillator Networks: Are Smaller Worlds Easier to Synchronize? Phys. Rev. Lett. **91**, 014101, (2003)
- NMP84. Novikov, S.P., Manakov, S.V., Pitaevskii, L.P., Zakharov, V.E.: Theory of Solitons, Plenum/Kluwer, Dordrecht, (1984)
- NNM00. Nagao, N., Nishimura, H., Matsui, N.: A Neural Chaos Model of Multistable Perception. Neural Processing Letters **12**(3): 267–276, (2000)
- NNM97. Nishimura, H., Nagao, N., Matsui, N.: A Perception Model of Ambiguous Figures based on the Neural Chaos, In: N. Kasabov et al. (eds): Progress in Connectionist-Based Information Systems, **1**, 89–92, Springer-Verlag, New York, (1997)
- NP77. Nicolis, G., Prigogine, I.: Self-Organization in Nonequilibrium Systems: From Dissipative Structures to Order through Fluctuations. Wiley Europe, (1977)
- NPT99. Nakamura, Y., Pashkin, Yu.A., Tsai, J.S.: Coherent control of macroscopic quantum states in a single-Cooper-pair box, Nature, **398**, 786–788, (1999)
- NS90. Nijmeijer, H., van der Schaft, A.J.: Nonlinear Dynamical Control Systems. Springer, New York, (1990)
- NSW01. Newman, M.E.J., Strogatz, S.H., Watts, D.J.: Random graphs with arbitrary degree distributions and their applications. Phys. Rev. E **64**, 026118, (2001)
- Nan95. Nanopoulos, D.V.: Theory of Brain Function, Quantum Mechanics and Superstrings. CERN-TH/95128, (1995)
- Nas83. Nash, C., Sen, S.: Topology and Geometry for Physicists. Academic Press, London, (1983)
- Nay73. Nayfeh, A.H.: Perturbation Methods. Wiley, New York, (1973)
- Neu58. Von Neumann, J.: The Computer and the Brain. Yale Univ. Press, (1958)
- New00. Newman, M.E.J.: Models of the small world. J. Stat. Phys. **101**, 819, (2000)
- New01. Newman, M.E.J.: Scientific collaboration networks. II. Shortest paths, weighted networks, and centrality. Phys. Rev. E **64**, 016132 (2001)
- New03. Newman, M.E.J.: Mixing patterns in networks. Phys. Rev. E **67**, 026126, (2003)
- New74. Newell, A.C. (ed): Nonlinear Wave Motion. Am. Mat. Soc., Providence, R.I., (1974)

- New80. Newhouse, S.: Lectures on dynamical systems. In *Dynamical Systems*, C.I.M.E. Lectures, 1–114, Birkhauser, Boston, MA, (1980)
- Nit00. Nitta, T.: An analysis on fundamental structure of complex-valued neuron. *Neu. Proc. Let.*, **12**(3), 239–246, (2000)
- Nit04. Nitta, T.: Reducibility of the Complex-valued Neural Network. *Neu. Inf. Proc.*, **2**(3), 53–56, (2004)
- Nit97. Nitta, T.: An extension of the back-propagation algorithm to complex numbers. *Neu. Net.*, **10**(8), 1392–1415, (1997)
- Noe99. Noest, J.: Discrete-state phasor neural networks. *Phys. Rev. A* **38**, 2196–2199, (1988)
- OBS92. Obermayer, K., Blasdel, G.G., Schulten, K.: *Phys. Rev. A* **45**, 7568–7589, (1992)
- OS74. Oldham, K.B., Spanier, J.: *The Fractional Calculus*. Academic Press, New York, (1974)
- OFS98. Okada, M., Fukai, T., Shiino, M.: Random and systematic dilutions of synaptic connections in a neural network with a nonmonotonic response function. *Phys. Rev. E* **57**, 2095–2103, (1998)
- OGY90. Ott, E., Grebogi, C., Yorke, J.A.: Controlling chaos. *Phys. Rev. Lett.*, **64**, 1196–1199, (1990)
- OK94. Oh, H., Kothari, S.C.: Adaptation of the Relaxation Method for Learning in Bidirectional Associative Memory. *IEEE Trans. Neu. Net.* **5**(4), 573–583, (1994)
- OS94. Ovchinnikov, Yu.N., Schmid, A.: Resonance phenomena in the current-voltage characteristic of a Josephson junction. *Phys. Rev. B* **50**, 63326339, (1994)
- OTK02. Ozbudak, E.M., Thattai, M., Kurtser, I., Grossman, A.D., van Oudenaarden, A.: Regulation of noise in the expression of a single gene. *Nature Gen.*, **31**, 69–73, (2002)
- OTL04. Ozbudak, E.M., Thattai, M., Lim, H.N., Shraiman, B.I., van Oudenaarden, A.: Multistability in the lactose utilization network of *Escherichia coli*. *Nature*, **427**(6976), 737–40, (2004)
- Oja82. Oja, E.: A simplified neuron modeled as a principal component analyzer. *J. Math. Biol.* **15**, 267–273, (1982)
- Oka95. Okada, M.: A hierarchy of macro-dynamical equations for associative memory. *Neural Networks* **8**, 833–838, (1995)
- Oka96. Okada, M.: Notions of associative memory and sparse coding. *Neu. Net.* **9**, 1429–1458, (1996)
- Omo86. Omohundro, S.M.: *Geometric Perturbation Theory in Physics*. World Scientific, Singapore, (1986)
- Ons35. Onsager, L.: Reciprocal relations in irreversible processes. II. *Phys. Rev.* **38**, 2265–2279, (1931)
- Ori01. Oriti, D.: Spacetime geometry from algebra: Spin foam models for non-perturbative quantum gravity. *Rept. Prog. Phys.* **64**, 1489–1544, (2001)
- Ose68. Oseledets, V.I.: A Multiplicative Ergodic Theorem: Characteristic Lyapunov Exponents of Dynamical Systems. *Trans. Moscow Math. Soc.*, **19**, 197–231, (1968)

- Ott89. Ottino, J.M.: The kinematics of mixing: stretching, chaos and transport. Cambridge Univ. Press, Cambridge, (1989)
- Ott93. Ott, E.: Chaos in dynamical systems. Cambridge University Press, Cambridge, (1993)
- PC89. Parker, T.S., Chua, L.O.: Practical Numerical Algorithm for Chaotic Systems, Springer-Verlag, New York, (1989)
- PB89. Peyrard, M., Bishop, A.R.: Statistical mechanics of a nonlinear model for DNA denaturation. Phys. Rev. Lett. **62**, 2755, (1989)
- PB94. Plischke, M. Bergersen, B.: Equilibrium Statistical Mechanics. World Scientific, Singapore, (1994)
- PB99. Paul, W., Baschnagel, J.: Stochastic Processes: from Physics to Finance. Springer, Berlin, (1999)
- PBK62. Pontryagin, L., Boltyanskii, V., Gamkrelidze, R., Mishchenko, E.: The mathematical theory of optimal processes. Interscience, (1962)
- PBR97. Prasad, A., Mebra, V., Ramaswamy, R.: Intermittency Route to Strange Nonchaotic Attractors. Phys. Rev. Lett. **79**, 41274130, (1997)
- PBS01. Potters, M., Bouchaud, J.P., Sestovic, D.: Hedged Monte-Carlo: low variance derivative pricing with objective probabilities. Physica A **289**, 517-25, (2001)
- PC90. Pecora, L.M., Carroll, T.L.: Synchronization in chaotic systems. Phys. Rev. Lett. **64**, 821-824, (1990)
- PC91. Pecora, L.M., Carroll, T.L.: Driving systems with chaotic signals. Phys. Rev. A **44**, 2374-2383, (1991)
- PC98. Pecora, L.M., Carroll, T.L.: Master stability functions for synchronized coupled systems. Phys. Rev. Lett. **80**, 2109-2112, (1998)
- PCS93. Penney, R.W., Coolen, A.C., Sherrington D.: Coupled dynamics of fast and slow interactions in neural networks and spin systems. J. Phys. A: Math. Gen. **26**, 3681, (1993)
- PEL00. Principe, J., Euliano, N., Lefebvre, C.: Neural and Adaptive Systems: Fundamentals Through Simulations. Wiley, New York, (2000)
- PHE92. Piaget, J., Henriques, G., Ascher, E.: Morphisms and categories. Erlbaum Associates, Hillsdale, NJ, (1992)
- PHV86. Posch, H.A., Hoover, W.G., Vesely, F.J.: Canonical Dynamics of the Nosé Oscillator: Stability, Order, and Chaos. Phys. Rev. A, **33**(6), 4253-4265, (1986)
- PI03. Pearce, C.E.M., Ivancevic, V.: A generalised Hamiltonian model for the dynamics of human motion. In Differential Equations and Applications, Vol. 2, Eds. Y.J. Cho, J.K. Kim and K.S. Ha, Nova Science, New York, (2003)
- PI04. Pearce, C.E.M., Ivancevic, V.: A qualitative Hamiltonian model for the dynamics of human motion. In Differential Equations and Applications, Vol. 3, Eds. Y.J. Cho, J.K. Kim and K.S. Ha, Nova Science, New York, (2004)

- PK97. Pikovsky, A., Kurth, J.: Coherence Resonance in a Noise-Driven Excitable Systems. *Phys. Rev. Lett.* **78**, 775-778, (1997)
- PLS00. Pappas, G.J., Lafferriere, G., Sastry, S.: Hierarchically consistent control systems. *IEEE Trans. Aut. Con.*, **45**(6), 1144-1160, (2000)
- PMZ02. Pitt, M.A., Myung, I.J., Zhang, S.: Toward a method of selecting among computational models of cognition *Psychological Review*, **109**(3), 472-491, (2002)
- POR97. Pikovsky, A., Osipov, G., Rosenblum, M., Zaks, M., Kurths, J.: Attractor-repeller collision and eyelet intermittency at the transition to phase synchronization. *Phys. Rev. Lett.* **79**, 47-50, (1997)
- PR03. Pedrycz, W., Reformat, M.: Evolutionary fuzzy modeling. *IEEE Trans. Fuzzy Syst.* **11**, 652-665, (2003)
- PR86. Penrose, R., Rindler, W.: *Spinors and Space-time II*. Cambr. Univ. Press, Cambridge, (1986)
- PRK01. Pikovsky, A., Rosenblum, M., Kurths, J.: *Synchronization: A Universal Concept in Nonlinear Sciences*. Cambridge Univ. Press, Cambridge, UK, (2001)
- PS62. Perring, J.K., Skyrme, T.R.H.: A model unified field equation. *Nuc. Phys.*, **31**, 550-555, (1962)
- PS89. Pugh, C.C., Shub, M.: Ergodic attractors. *Trans. Amer. Math. Soc.* **312**, 1-54, (1989)
- PS94. Penney, R.W., Sherrington, D.: Slow interaction dynamics in spin-glass models. *J. Phys. A: Math. Gen.* **27**, 4027-4041, (1994)
- PS97. Phillips, W.A., Singer, W.: In search of common foundations for cortical computation. *Behavioral and Brain Science*, **20**, 657-722, (1997)
- PS02. Pappas, G.J., Simic, S.: Consistent hierarchies of affine nonlinear systems. *IEEE Trans. Aut. Con.*, **47**(5), 745-756, (2002)
- PST86. Park, J.K., Steiglitz, K. and Thurston, W.P.: Soliton-like Behavior in Automata. *Physica D*, **19**, 423-432, (1986)
- PSV87. Paladin, G., Serva, M., Vulpiani, A.: Complexity in Dynamical Systems with Noise. *Phys. Rev. Lett.* **74**, 6669, (1995)
- PT93. Palis, J., Takens, F.: *Hyperbolicity and sensitive-chaotic dynamics at homoclinic bifurcations*. Cambridge University Press, (1993)
- PV03. Pessa, E., Vitiello, G.: Quantum noise, entanglement and chaos in the quantum field theory of mind/brain states, *Mind and Matter*, **1**, 59-79, (2003)
- PV04. Pessa, E., Vitiello, G.: Quantum noise induced entanglement and chaos in the dissipative quantum model of brain, *Int. J. Mod. Phys. B*, **18** 841-858, (2004)
- PV05. Potkonjak, V., Vukobratovic, M.: A generalized approach to modeling dynamics of human and humanoid motion, *Int. J. Adv. Robot. Sys.*, **2**(1), pp. 21-45, (2005)
- PV99. Pessa, E., Vitiello, G.: Quantum dissipation and neural net dynamics, *Bioelectrochem. Bioener.*, **48**, 339-342, (1999)

- PVB98. Prut, Y., Vaadia, E., Bergman, H., Haalman, I., Slovin, H., Abeles, M.: Spatiotemporal structure of cortical activity: properties and behavioral relevance. *J. Neurophysiol.* **79**, 2857–2874, (1998)
- PY97. Pan, S., Yin, F.: Optimal control of chaos with synchronization. *Int. J. Bif. Chaos* **7**, 2855–2860, (1997)
- PZR97. Pikovsky, A., Zaks, M., Rosenblum, M., Osipov, G., Kurths, J.: Phase synchronization of chaotic oscillations in terms of periodic orbits. *Chaos* **7**, 680, (1997)
- Pal88. Palmer, K.J.: Exponential dichotomies, the shadowing lemma and transversal homoclinic points, U. Kirchgraber and H.-O. Walther (eds), 265–306, *Dynamics Reported*, vol. I, Teubner-Wiley, Stuttgart/Chichester, (1988)
- Pal97. Palais, R.S.: The symmetries of solitons. *Bull. Amer. Math. Soc.* **34**, 339–403, (1997)
- Paz71. Paz, A.: *Introduction to Probabilistic Automata*. Academic, New York, (1971)
- Pee80. Peebles, P.J.E.: *The Large Scale Structure of the Universe*. Princeton Univ. Press, (1980)
- Pen67. Penrose, R.: Twistor algebra. *J. Math. Phys.*, **8**, 345–366, (1967)
- Pen89. Penrose, R.: *The Emperor's New Mind*, Oxford Univ. Press, Oxford, (1989)
- Pen94. Penrose, R.: *Shadows of the Mind*. Oxford Univ. Press, Oxford, (1994)
- Pen97. Penrose, R.: *The Large, the Small and the Human Mind*. Cambridge Univ. Press, (1997)
- Per84. Peretto, P.: Collective properties of neural networks: a statistical physics approach. *Biol. Cybern.* **50**, 51, (1984)
- Per92. Peretto, P.: *An Introduction to the Theory of Neural Computation*. Cambridge Univ. Press, Cambridge, (1992)
- Pes75. Peskin, C.: *Mathematical Aspects of Heart Physiology*. Courant Inst. Math. Sci., New York Univ, (1975)
- Pes76. Pesin, Ya.B.: Invariant manifold families which correspond to non-vanishing characteristic exponents. *Izv. Akad. Nauk SSSR Ser. Mat.* **40**(6), 1332–1379, (1976)
- Pes77. Pesin, Ya.B.: Lyapunov Characteristic Exponents and Smooth Ergodic Theory. *Russ. Math. Surveys*, **32**(4), 55–114, (1977)
- Pet02. Petras, I.: Control of Fractional-Order Chua's System. *J. El. Eng.* **53**(7-8), 219–222, (2002)
- Pet96. Petrov, V., Showalter, K.: Nonlinear Control from Time-Series. *Phys. Rev. Lett.* **76**, 3312, (1996)
- Pet98. Petersen, P.: *Riemannian Geometry*. Springer, New York, (1998)
- Pet99. Petersen, P.: Aspects of Global Riemannian Geometry. *Bull. Amer. Math. Soc.*, **36**(3), 297–344, (1999)
- Pet99. Petras, I.: The Fractional-order controllers: Methods for their synthesis and application. *J. El. Eng.* **9-10**, 284–288, (1999)

- Pic86. Pickover, C.A.: Computer Displays of Biological Forms Generated From Mathematical Feedback Loops. *Computer Graphics Forum*, **5**, 313, (1986)
- Pic87. Pickover, C.A.: Mathematics and Beauty: Time-Discrete Phase Planes Associated with the Cyclic System. *Computer Graphics Forum*, **11**, 217, (1987)
- Plu99. Plumbley, M.D.: Do cortical maps adapt to optimize information density?, *Network*, **10**, 41–58, (1999)
- Pod99. Podlubny, I.: *Fractional Differential Equations*. Academic Press, San Diego, (1999)
- Pol91. Pollack, J.B.: The induction of dynamical recognizers. *Mach. Learn.*, **7**, 227–252, (1991)
- Pop59. Popper, K.R.: *The Logic of Scientific Discovery*. New York, NY: Basic Books, (1959)
- Pos86. Postnikov, M.M.: *Lectures in Geometry V, Lie Groups and Lie Algebras*, Mir Publ., Moscow, (1986)
- Pra91. Pratt, V.: Modeling concurrency with geometry. in: *Proc. of the 18th ACM Symposium on Principles of Programming Languages*, (1991)
- Pri62. Prigogine, I.: *Introduction to thermodynamics of irreversible processes*. John Wiley, New York, (1962)
- Pul05. Pulvermüller, F.: Brain mechanisms kinking language and action. *Nature Rev. Neurosci.* **6**, 576–582, (2005)
- Put93. Puta, M.: *Hamiltonian Mechanical Systems and Geometric Quantization*, Kluwer, Dordrecht, (1993)
- Pyr92. Pyragas, K.: Continuous control of chaos, by self-controlling feedback. *Phys. Lett. A*, **170**, 421–428, (1992)
- Pyr95. Pyragas, K.: Control of chaos via extended delay feedback. *Phys. Lett. A*, **206**, 323–330, (1995)
- RBO87. Romeiras, J., Bodeson, A., Ott, E., Antonsen, T.M. Jr., Grebogi, C.: Quasiperiodically forced dynamical systems with strange nonchaotic attractors. *Physica D* **26**, 277–294, (1987)
- RBT01. Roe, R.M., Busemeyer, J.R., Townsend, J.T.: Multi-alternative decision field theory: A dynamic connectionist model of decision making. *Psych. Rev.*, **108**, 370–392, (2001)
- RD99. Reynolds, J.H., Desimone, R.: The role of neural mechanisms of attention to solve the binding problem. *Neuron*, **24**, 19–29, (1999)
- REB03. Rani, R., Edgar, S.B., Barnes, A.: Killing Tensors and Conformal Killing Tensors from Conformal Killing Vectors. *Class.Quant.Grav.*, **20**, 1929–1942, (2003)
- RG98. Rao, A.S., Georgeff, M.P.: Decision Procedures for BDI Logics. *Journal of Logic and Computation*, **8**(3), 292–343, (1998)
- RGL99. Rodriguez, E., George, N., Lachaux, J., Martinerie, J., Renault, B., Varela, F.: Long-distance synchronization of human brain activity. *Nature*, **397**, 430, (1999)
- RGO92. Romeiras, F.J., Grebogi, C., Ott, E., Dayawansa, W.P.: Using small perturbations to control chaos. *Physica D* **58**, 165–192, (1992)

- RH89. Rose, R.M., Hindmarsh, J.L.: The assembly of ionic currents in a thalamic neuron. I The three-dimensional model. *Proc. R. Soc. Lond. B*, **237**, 267–288, (1989)
- RKH88. Riedel, U., Kühn, R., van Hemmen, J.L.: Temporal sequences and chaos in neural nets. *Phys. Rev. A* **38**, 1105, (1988)
- RM86. Rumelhart, D. E., McClelland, J. L. and the PDP Research Group: *Parallel Distributed Processing*. vol. 1, MIT Press, Cambridge, MA, (1986)
- RMS90. Riani, M., Masulli, F., Simonotto, E.: Stochastic dynamics and input dimensionality in a two-layer neural network for modeling multistable perception. In: *Proc. IJCNN*. 1019–1022, (1990)
- RMS92. Ritter, H., Martinetz, T., Schulten, K.: *Neural Computation and Self-Organizing Maps: An Introduction*, Addison-Wesley, Reading, (1992)
- RMZ03. Racocanu, D., Minca, E., Zerhouni, N.: Fuzzy Petri nets for monitoring and recovery. In *Proc. 2003 IEEE Int. Conf. Robot. Automat.*, 4318–4323, Taipei, Taiwan, (2003)
- RN03. Russel, S., Norvig, P.: *Artificial Intelligence: A Modern Approach*. Prentice Hall, New Jersey, (2003)
- RO87. Romeiras, F.J., Ott, E.: Strange nonchaotic attractors of the damped pendulum with quasiperiodic forcing. *Phys. Rev. A* **35**, 44044413, (1987)
- ROH98. Rosa, E., Ott, E., Hess, M.H.: Transition to Phase Synchronization of Chaos. *Phys. Rev. Lett.* **80**, 1642–1645, (1998)
- RP99. Riesenhuber, M., Poggio, T.: Are cortical models really bound by the ‘Binding Problem’?. *Neuron*, **24**, 87–93, (1999)
- RPG94. Rasmussen, J., Pejtersen, A.M., Goodstein, L.P.: *Cognitive Systems Engineering*. Wiley, New York, (1994)
- RPK96. Rosenblum, M., Pikovsky, A., Kurths, J.: Phase synchronization of chaotic oscillators. *Phys. Rev. Lett.* **76**, 1804, (1996)
- RPK97. Rosenblum, M., Pikovsky, A., Kurths, J.: From Phase to Lag Synchronization in Coupled Chaotic Oscillators. *Phys. Rev. Lett.* **78**, 4193–4196, (1997)
- RS75. Reed, M., Simon, B.: *Methods of modern mathematical physics*, Vol. 2: Fourier analysis, self-adjointness. Academic Press, San Diego, (1975)
- RS86. Ritter, H., Schulten, K.: On the Stationary State of Kohonen’s Self-Organizing Sensory Mapping. *Biological Cybernetics*, **54**, 99–106, (1986)
- RS88. Ritter, H., Schulten, K.: Convergence Properties of Kohonen’s Topology Conserving Maps: Fluctuations, Stability and Dimension Selection. *Biological Cybernetics*, **60**, 59–71, (1988)
- RSW90. Richter, P.H., Scholz, H.-J., Wittek, A.: A breathing chaos. *Nonlinearity* **3**, 45–67, (1990)
- RT02. Rosa-Clot, M., Taddei, S.: A Path Integral Approach to Derivative Security Pricing. *Int. J. Theor. Appl. Finance*, **5**(2), 123–146, (2002)
- RT71. Ruelle, D., Takens, F.: On the nature of turbulence. *Comm. Math. Phys.*, **20**, 167–192, (1971)

- RT90. Reshetikhin, N., Turaev, V.: Ribbon graphs and their invariants derived from quantum groups. *Comm. Math. Phys.* **127**, 1–26, (1990)
- RU67. Ricciardi, L.M., Umezawa, H.: Brain physics and many-body problems, *Kibernetik*, **4**, 44, (1967)
- RW87. Ramadge, P.J., Wonham, W.M.: Supervisory control of a class of discrete event processes. *SIAM J. Control Optim.* **25**, 206–230, (1987)
- RYD96. Ryu, S., Yu, W., Stroud, D.: Dynamics of an underdamped Josephson-junction ladder. *Phys. Rev. E* **53**, 2190–2195, (1996)
- Raj82. Rajaraman, R.: Solitons and Instantons. North-Holland, Amsterdam, (1982)
- Ram90. Ramond, P.: *Field Theory: a Modern Primer*. Addison-Wesley, Reading, MA, (1990)
- Ras67. Rashevsky, N.: Organismic sets. Outline of a general theory of biological and sociological organisms. *Bull. Math. Biophys.*, **29**, 139–152, (1967)
- Rat69. Ratner, M.: Markov partitions for Anosov flows on 3-dimensional manifolds. *Mat. Zam.* **6**, 693–704, (1969)
- Rat78. Ratcliff, R.: A theory of memory retrieval. *Psych. Rev.*, **85**, 59–108, (1978)
- Reg61. Regge, T.: General relativity without coordinates. *Nuovo Cim. A* **19**, 558–571, (1961)
- Rei48. Reidemeister, K.: *Knotentheorie*, Chelsea Pub., New York, (1948)
- Rei98. Reimann, S.: On the design of artificial auto-associative neural networks. *Neu. Net.* **11**, 611–621, (1998)
- Rei98. O'Reilly, R.C.: Six principles for biologically based computational models of cortical cognition. *Trends Cog. Sci.* **2**, 455–462, (1998)
- Rey03. Reyes, A.: Synchrony-dependent propagation of firing rate in iteratively constructed networks in vitro. *Nature Neurosci.* **6**, 593–599, (2003)
- Rha84. de Rham, G.: *Differentiable Manifolds*. Springer, Berlin, (1984)
- Ric93. Ricca, R.L.: Torus knots and polynomial invariants for a class of soliton equations, *Chaos*, **3**(1), 83–91, (1993)
- Rie90. Rieger, H.: Storing an extensive number of grey-toned patterns in a neural network using multi-state neurons. *J. Phys. A* **23**, L1273–L1279, (1990)
- Rig03. Rigatos, G.G.: Fuzzy stochastic automata for intelligent vehicle control. *IEEE Trans. Indust. Elect.* **50**, 76–79, (2003)
- Rob79. Robbins, K.: Periodic solutions and bifurcation structure at high r in the Lorenz system. *SIAM J. Appl. Math.* **36**, 457–472, (1979)
- Rob95. Robinson, C.: *Dynamical Systems*. CRC Press. Boca Raton, FL, (1995)

- Ros58a. Rosen, R.: (a) The representation of biological systems from the standpoint of the theory of categories. *Bull. Math. Biophys.*, **20**, 245–260; (b) A relational theory of biological systems. *Bull. Math. Biophys.*, **20**, 317–341, (1958)
- Ros58b. Rosenblatt F.: The perceptron: a probabilistic model for information storage and organization in the brain. *Physiol. Rev.*, **65**, 386–408, (1958)
- Ros76. Rössler, O.E.: An Equation for Continuous Chaos. *Phys. Lett. A*, **57**(5), 397–398, (1976)
- Rov98. Rovelli, C.: Loop quantum gravity. *Living Rev. Rel.* **1**, (1998)
- Rus844. Russell, J.S.: Report on Waves, 14th meeting of the British Association for the Advancement of Science, BAAS, London, (1844)
- Rus885. Russell, J.S.: The Wave of Translation in the Oceans of Water, Air and Ether. Trübner, London, (1885)
- Rue76. Ruelle, D.: A measure associated with Axiom A attractors. *Am. J. Math.* **98**, 619–654, (1976)
- Rue78a. Ruelle, D.: An inequality for the entropy of differentiable maps. *Bol. Soc. Bras. Mat.* **9**, 83–87, (1978)
- Rue78b. Ruelle, D.: Thermodynamic formalism. Addison-Wesley, Reading, MA, (1978)
- Rue78c. Ruelle, D.: Thermodynamic formalism. *Encyclopaedia of Mathematics and its Applications*. Addison-Wesley, Reading, (1978)
- Rue79. Ruelle, D.: Ergodic theory of differentiable dynamical systems. *Publ. Math. IHES* **50**, 27–58, (1979)
- Rue89. Ruelle, D.: The thermodynamical formalism for expanding maps. *Commun. Math. Phys.* **125**, 239–262, (1989)
- Rue98. Ruelle, D.: Nonequilibrium statistical mechanics near equilibrium: computing higher order terms. *Nonlinearity* **11**, 5–18, (1998)
- Rue99. Ruelle, D.: Smooth Dynamics and New Theoretical Ideas in Non-equilibrium Statistical Mechanics. *J. Stat. Phys.* **95**, 393–468, (1999)
- Rul01. Rulkov, N.F.: Regularization of Synchronized Chaotic Bursts. *Phys. Rev. Lett.* **86**, 183–186, (2001)
- Ryd96. Ryder, L.: Quantum Field Theory. Cambridge Univ. Press, (1996)
- SA98. Schaal, S., Atkeson, C.G.: Constructive incremental learning from only local information. *Neural Comput.*, **10**, 2047–2084, (1998)
- SB74. Sayers, B.McA., Beagley, H.A.: Objective evaluation of auditory evoked EEG responses. *Nature*, **251**, 608–609, (1974)
- SB89. Shapiro, M., Brumer, P.: Coherent Radiative Control of Unimolecular Reactions: Selective Bond Breaking with Picosecond Pulses. *J. Chem. Phys.* **90**, 6179, (1989)
- SB98. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge, MA, (1998)
- SC00. Skantzos, N.S., Coolen, A.C.C.: $(1 + \infty)$ -Dimensional Attractor Neural Networks. *J. Phys. A* **33**, 5785–5807, (2000)

- SC01. Segel, L.A., Cohen, I.R.(eds.): Design principles for the immune system and other distributed autonomous systems, Oxford Univ. Press, (2001)
- SC01. Skantzos, N.S., Coolen, A.C.C.: Attractor Modulation and Proliferation in $(1 + \infty)$ -Dimensional Neural Networks. *J. Phys. A* **34**, 929-942, (2001)
- Sch78. Schot, S.H.: Jerk: The Time Rate of Change of Acceleration. *Am. J. Phys.* **46**(11), 1090-1094, (1978)
- SDK53. Seeger, A., Donth, H., Kochendörfer, A.: Theorie der Versetzungen in eindimensionalen Atomreihen. *Zeitschrift für Physik*, **134**, 173-93 (1953)
- SE04. Sun, Y.L., Er, M.J.: Hybrid fuzzy control of robotic systems. *IEEE Trans. Fuz. Sys.* **12**(6), 755-765, (2004)
- SF92. Shiino, M., Fukai, T.: Self-Consistent Signal-to-Noise Analysis and its Application to Analogue Neural Networks with Asymmetric Connections. *J. Phys. A: Math. Gen.* **25**, L375-L381, (1992)
- SFM98. Skiffington, S., Fernandez, E., McFarland, K.: Towards a validation of multiple features in the assessment of emotions. *Eur. J. Psych. Asses.* **14**(3), (1998)
- SG95. Singer, W., Gray, C.M.: Visual feature integration and temporal correlation hypothesis. *Ann. Rev. Neurosci.*, **18**, 555-586, (1995)
- SGL93. Shih, C.L., Gruver, W. Lee, T.: Inverse kinematics and inverse dynamics for control of a biped walking machine. *J. Robot. Syst.*, **10**, 531-555, (1993)
- SHT02. Shimizu-Sato, S., Huq, E., Tepperman, J.M., Quail, P.H.: A light-switchable gene promoter system. *Nature, Biotech.*, **20**, 1041-1044, (2002)
- SI89. Sastry, S.S., Isidori, A.: Adaptive control of linearizable systems. *IEEE Trans. Aut. Con.*, **34**(11), 1123-1131, (1989)
- SIT95. Shea, H.R., Itzler, M.A., Tinkham, M.: Inductance effects and dimensionality crossover in hybrid superconducting arrays *Phys. Rev. B* **51**, 1269012697, (1995)
- SJD94. Schiff, S.J., Jerger, K., Duong, D.H., Chang, T., Spano, M.L., Ditto, W.L.: Controlling chaos in the brain. *Nature*, **370**, 615-620, (1994)
- SJK97. Schoenauer, M., Jouve, F., Kallel, L.: Identification of mechanical inclusions. In D.Dasgupta and Z. Michalewicz, editors, *Evolutionary Algorithms in Engineering Applications*. Springer Verlag, (1997)
- SK75. Sherrington, D., Kirkpatrick S.: Solvable Model of a Spin-Glass. *Phys. Rev. Lett.* **35**, 17921796, (1975)
- SK86. Sompolinsky, H., Kanter, I.: Temporal Association in Asymmetric Neural Networks. *Phys. Rev. Lett.* **57**, 2861, (1986)
- SK93. Shih, C.L., Klein, C.A.: An adaptive gait for legged walking machines over rough terrain. *IEEE Trans. Syst. Man, Cyber. A*, **23**, 1150-1154, (1993)

- SK98. Shabanov, S.V., Klauder, J.R.: Path Integral Quantization and Riemannian-Symplectic Manifolds. *Phys. Lett. B* **435**, 343-349, (1998)
- SKM93. Samko, S.G., Kilbas, A.A., Marichev, O.I.: Fractional Integrals and Derivatives Theory and Applications. Gordon & Breach, New York, (1993)
- SKO04. Shouno, M., Kido, S., Okada, M.: Analysis of Bidirectional Associative Memory using Self-consistent Signal to Noise Analysis and Statistical Neurodynamics. *J. Phys. Soc. Jpn.* **73**(9), 2406-2412, (2004)
- SKW95. Sakai, K., Katayama, T., Wada, S., Oiwa, K.: Chaos causes perspective reversals for ambiguous patterns. In: *Advances in Intelligent Computing IPMU'94*, 463-472, (1995)
- SL00. Sprott, J.C., Linz, S.J.: Algebraically Simple Chaotic Flows. *Int. J. Chaos Theory and Appl.*, **5**(2), 3-22, (2000)
- SL03. Stevens, B.L., Lewis, F.L.: Aircraft Control and Simulation (2nd ed.). Wiley, Hoboken, NJ, (2003)
- SLR02. Sánchez, A., López, M.J., Rodríguez, M.A.: Nonequilibrium Phase Transitions in Directed Small-World Networks. *Phys. Rev. Lett.* **88**, 048701, (2002)
- SN79. Shimada, I., Nagashima, T.: A Numerical Approach to Ergodic Problem of Dissipative Dynamical Systems. *Prog. The. Phys.* **61**(6), 1605-16, (1979)
- SOG90. Shinbrot, T., Ott, E., Grebogi, C., Yorke, J.A.: Using chaos to direct trajectories to targets. *Phys. Rev. Lett.* **65**, 32153218, (1990)
- SR90. Shi, S., Rabitz, H.: Quantum mechanical optimal control of physical observables in microsystems. *J. Chem. Phys.* **92**(1), 364-376, (1990)
- SR91. Schwieters, C.D., Rabitz, H.: Optimal control of nonlinear classical systems with application to unimolecular dissociation reactions and chaotic potentials. *Phys. Rev. A* **44**, 52245238, (1991)
- SR91. Shi, S., Rabitz, H.: Optimal Control of Bond Selectivity in Unimolecular Reactions. *Comp. Phys. Com.* **63**, 71, (1991)
- SRK98. C. Schäfer, M.G. Rosenblum, J. Kurths, and H.-H. Abel: Heartbeat Synchronized with Ventilation. *Nature* **392**, 239-240 (1998)
- SRV04. Stoico, C., Renzi, D., Vericat, F.: From Darwin to Sommerfeld: Genetic algorithms and the electron gas. *arXiv:cond-mat/0412239*, (2004)
- SS01. Scholkopf, B., Smola, A.: *Learning with Kernels*. MIT Press, Cambridge, MA, (2001)
- SSK87. Sakaguchi, H., Shinomoto, S., Kuramoto, Y.: Local and global self-entrainments in oscillator-lattices. *Prog. Theor. Phys.* **77**, 1005-1010, (1987)
- SSP96. Schwenker, F., Sommer, F.T., Palm, G.: Iterative retrieval of sparsely coded associative memory patterns. *Neu. Net.* **9**, 445-455, (1996)

- SSR93. Shen, L., Shi, S., Rabitz, H., Lin, C., Littman, M., Heritage, J.P., Weiner, A.M.: Optimal control of the electric susceptibility of a molecular gas by designed nonresonant laser pulses of limited amplitude. *J. Chem. Phys.* **98**, 7792-7803, (1993)
- ST98. Schultz, S., Treves, A.: Stability of the replica-symmetric solution for the information conveyed by a neural network. *Phys. Rev. E* **57**, 3302-3310, (1998)
- STP95. Schaub, H., Tsiotras, P., Junkins, J.: Principal Rotation Representations of Proper $N \times N$ Orthogonal Matrices. *Int. J. Eng. Sci.*, **33**, (15), 2277-2295, (1995)
- STU78. Stuart, C.I.J., Takahashi, Y., Umezawa, H.: On the stability and non-local properties of memory, *J. Theor. Biol.* **71**, 605-618, (1978)
- STU79. Stuart, C.I.J., Takahashi, Y., Umezawa, H.: Mixed system brain dynamics: neural memory as a macroscopic ordered state, *Found. Phys.* **9**, 301, (1979)
- STZ93. Satarić, M.V., Tuszynski, J.A., Zakula, R.B.: Kinklike excitations as an energy-transfer mechanism in microtubules. *Phys. Rev. E*, **48**, 589-597, (1993)
- SV96. Strominger, A., Vafa, C.: On the microscopic origin of the Bekenstein-Hawking entropy. *Phys. Lett. B3* **79**, 99, (1996)
- SVW95. Srivastava, Y.N., Vitiello, G., Widom, A.: Quantum dissipation and quantum noise, *Annals Phys.*, **238**, 200, (1995)
- SW72. Sulanke R., Wintgen P.: Differential geometry und faserbündel; bound 75, Veb. Deutscher Verlag der Wissenschaften, Berlin, (1972)
- SW87. Shapere, A. Wilczek, F.: Two Applications of Axion Electrodynamics. *Phys. Rev. Lett.* **58**, 2051, (1987)
- SW89. Shapere, A. Wilczek, F.: Geometric Phases in Physics. World Scientific, Singapore, (1989)
- SW90. Steinlein, H., Walther, H.-O.: Hyperbolic Sets, Transversal Homoclinic Trajectories, and Symbolic Dynamics for C^1 -maps in Banach Spaces. *J. Dyn. Diff. Eqs.* **2**, 325-365, (1990)
- SW90. Sassetti, M., Weiss, U.: Universality in the dissipative two-state system. *Phys. Rev. Lett.* **65**, 2262-2265, (1990)
- SWB97. Shim, G.M., Wong, K.Y.M., Bollé, D. Dynamics of temporal activity in multi-state neural networks. *J. Phys. A*, **30**, 2637-2652, (1997)
- SWV99. Srivastava, Y.N., Widom, A., Vitiello, G.: Quantum measurements, information and entropy production. *Int. J. Mod. Phys. B13*, 3369-3382, (1999)
- SZT98. Satarić, M.V., Zeković, S., Tuszynski, J.A., Pokorný, J.: Mössbauer effect as a possible tool in detecting nonlinear excitations in microtubules. *Phys. Rev. E* **58**, 6333-6339, (1998)
- SZZ98. Shi, H., Zhao, Y., Zhuang, X.: A General Model for Bidirectional Associative Memories. *IEEE Trans. on Systems, Man and Cybernetics-Part B: Cybernetics*, **28**(4), 511-519, (1998)
- San68. Santos, E.S.: Maxmin automata. *Inform. Contr.* **13**, 363-377, (1968)

- Sch01. Schuster, H.G.: Complex Adaptive Systems: An Introduction. Scator, (2001)
- Sch81. Schulman, L.S.: Techniques and Applications of Path Integration, John Wiley & Sons, New York, (1981)
- Sch88. Schuster, H.G.(ed): Handbook of Chaos Control. Wiley-VCH, (1999)
- Sch91. Schroeder, M.: Fractals, Chaos, Power Laws. Freeman, New York, (1991)
- Sch98. Schaal, S.: Robot learning. In M. Arbib (ed). Handbook of Brain Theory and Neural Networks (2nd ed.), MIT Press, Cambridge, (1998)
- Sch99. Schaal, S.: Is imitation learning the route to humanoid robots?. Trends Cogn. Sci., **3**, 233–242, (1999)
- Sco04. Scott, A. (ed): Encyclopedia of Nonlinear Science. Routledge, New York, (2004)
- Sco99. Scott, A.C.: Nonlinear Science: Emergence and Dynamics of Coherent Structures, Oxford Univ. Press, Oxford, (1999)
- Sei95. Seiler, W.M.: Involution and constrained dynamics II: The Faddeev-Jackiw approach, J. Phys. A, **28**, 7315–7331, (1995)
- Sha06. Sharma, S.: An Exploratory Study of Chaos in Human-Machine System Dynamics, IEEE Trans. SMC B. 36(2), 319–326, (2006)
- Sha48. Shannon, C.E.: A mathematical theory for communication. Bell Sys. Tec. J. **27**, 379, (1948)
- She05. Shen, Z.: Riemann–Finsler geometry with applications to information geometry, unpublished material, (2005)
- Shi87. Shinomoto, S.: Memory maintenance in neural networks. J. Phys. A: Math. Gen. **20**, L1305–L1309, (1987)
- Shi98. Shishikura, M.: The Hausdorff dimension of the boundary of the Mandelbrot set and Julia sets. Ann. of Math. **147**, 225–267, (1998)
- Shu93. Shuster, M.D.: A Survey of Attitude Representations, J. Astr. Sci., **41**(4), 316–321, (1993)
- Sim90. Simpson, P.K.: Higher-Ordered and Intraconnected Bidirectional Associative Memories. IEEE Trans. Sys. Man Cyb. **20**(3), 637–652, (1990)
- Sin68a. Sinai, Ya.G.: Markov partitions and C-diffeomorphisms. Funkts. Analiz i Ego Pril. **2**(1), 64–89, (1968)
- Sin68b. Sinai, Ya.G.: Constuction of Markov partitions. Funkts. Analiz i Ego Pril. **2**(3), 70–80, (1968)
- Sin72. Sinai, Ya.G.: Gibbsian measures in ergodic theory. Uspehi Mat. Nauk **27**(4), 21–64, (1972)
- Sin89. Sinai, Ya.G.: Dynamical Systems II, Encyclopaedia of Mathematical Sciences, **2**, Springer-Verlag, Berlin, (1989)
- Sio05. Sioutis, C.: Reasoning and learning for intelligent agents. PhD thesis, Univ. SA, Adelaide, SA, (2005)
- Siv77. Sivashinsky, G.I.: Nonlinear analysis of hydrodynamical instability in laminar flames – I. Derivation of basic equations. Acta Astr. **4**, 1177, (1977)

- Sm88. Strogatz, S.H., Mirollo, R.E.: Phase-locking and critical phenomena in lattices of coupled nonlinear oscillators with random intrinsic frequencies. *Physica D* **31**, 143, (1988)
- Sma60. Smale, S.: The generalized Poincaré conjecture in higher dimensions, *Bull. Amer. Math. Soc.*, **66**, 373–375, (1960)
- Sma67. Smale, S.: Differentiable dynamical systems, *Bull. Amer. Math. Soc.*, **73**, 747–817, (1967)
- Sma67. Smale, S.: Differentiable Dynamical Systems. *Bull. AMS* **73**, 747–817, (1967)
- Sma99. van der Smagt, P.: (ed.) Self-Learning Robots. Workshop: Brainstyle Robotics, IEE, London, (1999)
- Sni80. Sniatycki, J.: Geometric Quantization and Quantum Mechanics. Springer-Verlag, Berlin, (1980)
- Soc91. Socolovsky, M.: Gauge transformations in fibre bundle theory. *J. Math. Phys.*, **32**, 2522, (1991)
- Spa82. Sparrow, C.: The Lorenz Equations: Bifurcations, Chaos, and Strange Attractors. Springer, New York, (1982)
- Spr93a. Sprott, J.C.: Automatic Generation of Strange Attractors. *Comput. & Graphics*, **17**(3), 325–332, (1993)
- Spr93b. Sprott, J.C.: Strange Attractors: Creating Patterns in Chaos. M&T Books, New York, (1993)
- Spr94. Sprott, J.C.: Some Simple Chaotic Flows. *Phys. Rev. E*, **50**(2), R647–R650, (1994)
- Spr97. Sprott, J.C.: Some Simple Chaotic Jerk Functions. *Am. J. Phys.*, **65**(6), 537–543, (1997)
- Sta00. Stanislavsky, A.A.: Memory effects and macroscopic manifestation of randomness. *Phys. Rev. E* **61**, 4752, (2000)
- Sta63. Stasheff, J.D.: Homotopy associativity of H -spaces I & II. *Trans. Amer. Math. Soc.*, **108**, 275–292, 293–312, (1963)
- Sta93. Stapp, H.P.: Mind, Matter and Quantum Mechanics. Springer, New York, (1993)
- Sta97. Stark, J.: Invariant Graphs for Forced Systems. *Physica D* **109**, 163–179, (1997)
- Ste36. Steuerwald, R.: Über Enneper'sche Flächen und Bäcklund'sche Transformation. *Abhandlungen der Bayerischen Akademie der Wissenschaften München*, 1–105, (1936)
- Ste69. Sternberg, S.: Memory-scanning: Mental processes revealed by reaction-time experiments. *Am. Sci.*, **57**(4), 421–457, (1969)
- Ste72. Steenrod, N.: The Topology of Fibre Bundles, Princeton Univ. Press, Princeton, (1972)
- Ste92. Stein, D.L.: Spin Glasses in Biology. World Scientific, Singapore, (1992)
- Ste93. Stengel, R.: Optimal control and estimation. Dover, New York, (1993)
- Sto68. Stong, R.E.: Notes on Cobordism Theory. Princeton Univ. Press, Princeton, (1968)
- Str00. Strogatz, S.H.: From Kuramoto to Crawford: exploring the onset of synchronization in populations of coupled oscillators. *Physica D*, **143**, 1–20, (2000)

- Str01. Strogatz, S.H.: Exploring complex networks. *Nature*, **410**, 268, (2001)
- Str68. Strang, G.: On the Construction and Comparison of Difference Schemes. *SIAM J. Num. Anal.* **5**, 506–517, (1968)
- Str94. Strogatz, S.: *Nonlinear Dynamics and Chaos*. Addison-Wesley, Reading, MA, (1994)
- Stu99. Stuart, J.: *Calculus* (4th ed.). Brooks/Cole Publ., Pacific Grove, CA, (1999)
- Sus83. Sussmann, H.J.: Lie brackets and local controllability: a sufficient condition for scalar-input systems, *SIAM J. Con. Opt.*, **21**(5), 686–713, (1983)
- Sus87. Sussmann, H.J.: A general theorem on local controllability, *SIAM J. Con. Opt.*, **25**(1), 158–194, (1987)
- Swi75. Switzer, R.K.: *Algebraic Topology – Homology and Homotopy*. (in *Classics in Mathematics*), Springer, New York, (1975)
- TF91. Tsue, Y., Fujiwara, Y.: Time-Dependent Variational Approach to (1+1)-Dimensional Scalar-Field Solitons, *Progr. Theor. Phys.* **86**(2), 469–489, (1991)
- TI99. Taiji, M., Ikegami, T.: Dynamics of internal models in game players. *Physica D*, **134**, 253–266, (1999)
- TKK00. Tanaka, T., Kakiya, S., Kabashima, Y.: Capacity Analysis of Bidirectional Associative Memory. In *Proc. Seventh Int. Conf. Neural Information Processing*, Vol 2, 779–784, Taejon, Korea, (2000)
- TLO97. Tanaka, H.A., Lichtenberg, A.J., Oishi, S.: Self-synchronization of coupled oscillators. with hysteretic response. *Physica D* **100**, 279, (1997)
- TLO97. Tanaka, H.A., Lichtenberg, A.J., Oishi, S.: First Order Phase Transition Resulting from Finite Inertia in Coupled Oscillator Systems. *Phys. Rev. Lett.* **78**, 21042107, (1997)
- TLZ02. Tang, X., Lou, S., Zhang, Y.: Chaos in soliton systems and special Lax pairs for chaos systems. Localized excitations in (2+ 1)-dimensional systems. *Phys. Rev. E* **66**, 046601, (2002)
- TM01. Trees, B.R., Murgescu, R.A.: Phase locking in Josephson ladders and the discrete sine-Gordon equation: The effects of boundary conditions and current-induced magnetic fields. *Phys. Rev. E* **64**, 046205, (2001)
- TMO00. Tras, E., Mazo, J.J., Orlando, T.P.: Discrete Breathers in Non-linear Lattices: Experimental Detection in a Josephson Array. *Phys. Rev. Lett.* **84**, 741, (2000)
- TP01. Tabuada, P., Pappas, G.J.: Abstractions of Hamiltonian Control Systems. *Proceedings of the 40th IEEE Conf. Decis. Con.*, Orlando, FL, (2001)
- TPS98. Tomlin, C., Pappas, G.J., Sastry, S.: Conflict resolution for air traffic management: A case study in multi-agent hybrid systems, *IEEE Trans. Aut. Con.*, **43**, 509–521, (1998)
- TR85. Tannor, D.J., Rice, S.A.: Control of selectivity of chemical reaction via control of wave packet evolution. *J. Chem. Phys.* **83**, 5013–5018, (1985)

- TRW98. Tass, P., Rosenblum, M.G., Weule, J., Kurths, J. *et al.*: Detection of $n : m$ Phase Locking from Noisy Data: Application to Magnetoencephalography. *Phys. Rev. Lett.* **81**, 3291–3294, (1998)
- TS01. Thompson, J.M.T., Stewart, H.B.: *Nonlinear Dynamics and Chaos: Geometrical Methods for Engineers and Scientists*. Wiley, New York, (2001)
- TS85. Takagi, T., Sugeno, M.: Fuzzy Identification of Systems and its Application to Modeling and Control. *IEEE Trans. Syst., Man, Cybern.* **15**, 116–132, (1985)
- TS99. Trees, B.R., Stroud, D.: Two-dimensional arrays of Josephson junctions in a magnetic field: A stability analysis of synchronized states. *Phys. Rev. B* **59**, 71087115, (1999)
- TSS05. Trees, B.R., Saranathan, V., Stroud, D.: Synchronization in disordered Josephson junction arrays: Small-world connections and the Kuramoto model. *Phys. Rev. E* **71**, 016215, (2005)
- TT04. Teramae, J.N., Tanaka, D.: Robustness of the noise-induced phase synchronization in a general class of limit cycle oscillators. *Phys. Rev. Lett.* **93**, 204103, (2004)
- TVP99. Tabony, J., Vuillard, L., Papaseit, C.: Biological self-organisation and pattern formation by way of microtubule reaction-diffusion processes. *Adv. Complex Syst.* **2**(3), 221–276, (1999)
- TW01. Tanaka, K., Wang, H.O.: *Fuzzy Control Systems Design and Analysis*. Wiley, New York, (2001)
- TW95. Tucker, R., Wang, C.: Black holes with Weyl charge and non-Riemannian waves. *Class. Quant. Grav.* **12**, 2587, (1995)
- TWG02. Timme, M., Wolf, F., Geisel, T.: Prevalence of unstable attractors in networks of pulse-coupled oscillators. *Phys. Rev. Lett.* **89**, 154105, (2002)
- TWG03. Timme, M., Wolf, F., Geisel, T.: Unstable attractors induce perpetual synchronization and desynchronization. *Chaos*, **13**, 377, (2003)
- TZ05. Tarasov, V.E., Zaslavsky, G.M.: Fractional Ginzburg-Landau equation for fractal media. *Physica A* **354**, 249–261, (2005)
- Tar04a. Tarasov, V.E.: Fractional Generalization of Liouville Equation. *Chaos* **14**, 123–127, (2004)
- Tar04b. Tarasov, V.E.: Path integral for quantum operations. *J. Phys. A* **37**, 3241–3257, (2004)
- Tar05a. Tarasov, V.E.: Fractional Systems and Fractional Bogoliubov Hierarchy Equations. *Phys. Rev. E* **71**, 011102, (2005)
- Tar05b. Tarasov, V.E.: Continuous Medium Model for Fractal Media. *Phys. Lett. A* **336**, 167–174, (2005)
- Tar05c. Tarasov, V.E.: Fractional Fokker-Planck Equation for Fractal Media. *Chaos* **15**, 023102, (2005)
- Tar05d. Tarasov, V.E.: Fractional Generalization of Gradient and Hamiltonian Systems. *J. Phys. A* **38**(26), 5929–5943, (2005)
- Tay21. Taylor, G.I.: Diffusion by continuous movements. *Proc. London Math. Soc.* **20**, 196, (1921)

- Tei83. Teitelboim, C.: Causality versus gauge invariance in quantum gravity and supergravity. *Phys. Rev. Lett.* **50**, 705–708, (1983)
- Thi79. Thirring, W.: *A Course in Mathematical Physics* (in four volumes). Springer, New York, (1979)
- Tho75. Thom, R.: *Structural Stability and Morphogenesis*. Addison–Wesley, Reading, (1975)
- Tho79. Thorpe, J.A.: *Elementary Topics in Differential Geometry*. Springer, New York, (1979)
- Tho93. Thompson, G.: Non-uniqueness of metrics compatible with a symmetric connection. *Class. Quant. Grav.* **10**, 2035, (1993)
- Tod67. Toda, M.: Vibration of a chain with nonlinear interactions. *J. Phys. Soc. Jap.* **22**, 431–36; also, Wave propagation in anharmonic lattices. *J. Phys. Soc. Jap.* **23**, 501–06, (1967)
- UC02. Uezu, T., Coolen, A.C.C.: Hierarchical Self-Programming in Recurrent Neural Networks. *J. Phys. A* **35**, 2761–2809, (2002)
- UMM93. Ustinov, A.V., Cirillo, M., Malomed, B.A.: Fluxon dynamics in one-dimensional Josephson-junction arrays. *Phys. Rev. B* **47**, 83578360, (1993)
- UN99. Udriste, C., Neagu, M.: Extrema of p -energy functional on a Finsler manifold. *Diff. Geom. Dyn. Sys.* 1(1), 10–19, (1999)
- Udr00. Udriste, C.: *Geometric dynamics*. Kluwer Academic Publishers, (2000)
- Ush99. Ushio, T.: Synthesis of Synchronized Chaotic Systems Based on Observers. *Int. J. Bif. Chaos* **9**, 541–546, (1999)
- VKD01. Van der Kooij, H., Jacobs R., van der Helm F.: An adaptive model of sensory integration in a dynamic environment applied to human stance control. *Biol. Cybern* **84**, 103–115, (2001)
- VKD03. Van der Kooij, H., Donker, S.: Use of Adaptive Model of Balance Control in the Identification of Postural Dynamics. *Proceedings of ISB'03*, Univ. Otago, Dunedin, NZ, (2003)
- Vai94. Vaisman, I.: *Lectures on the Geometry of Poisson Manifolds*. Birkhäuser Verlag, Basel, (1994)
- Vap95. Vapnik, V.: *The Nature of Statistical Learning Theory*. Springer, New York, (1995)
- Vap98. Vapnik, V.: *Statistical Learning Theory*. Wiley, New York, (1998)
- Via97. Viana, M.: Multidimensional nonhyperbolic attractors. *Publ. Math. IHES* **85**, 63–96, (1997)
- Vit01. Vitiello, G.: *My Double Unveiled*. John Benjamins, Amsterdam, (2001)
- Vit95. Vitiello, G.: Dissipation and memory capacity in the quantum brain model, *Int. J. Mod. Phys. B*, **9**, 973–989, (1995)
- Voi02. Voisin, C.: *Hodge Theory and Complex Algebraic Geometry I*. Cambridge Univ. Press, Cambridge, (2002)
- Vre96. Van Vreeswijk, C.: Partial synchronization in populations of pulse-coupled oscillators. *Phys. Rev. E* **54**, 5522, (1996)
- Vyg82. Vygotsky, L.S. *Historical meaning of the Psychological crisis. Collected works. Vol. 1. Pedag. Publ., Moscow*, (1982)

- WA00. Wall, M.M., Amemiya, Y.: Estimation for polynomial structural equation models. *Journal of American Statistical Association*, **95**, 929–940, (2000)
- WA98. Wall, M.M., Amemiya, Y.: Fitting nonlinear structural equation models. *Proc. Social Stat. Section. Ann. Meet. Ame. Stat. Assoc.* 180–185, (1998)
- WB95. Welch, G., Bishop, G.: An Introduction to the Kalman Filter. Univ. North Carolina, Dep. Comp. Sci., TR 95–041, (1995)
- WBB94. Wiesenfeld, K., Benz, S.P., Booii, P.A.: Phase-locked oscillator optimization for arrays of Josephson junctions. *J. Appl. Phys.* **76**, 3835–3846, (1994)
- WBL69. Willshaw, D.J., Buneman, O.P., Longuet-Higgins, H.C.: Non-holographic associative memory. *Nature*, **222**, 960–962, (1969)
- WC02. Wang, X.F., Chen, G.: Synchronization in small-world dynamical networks. *Int. J. Bifur. Chaos* **12**, 187, (2002)
- WCL96. Whan, C.B., Cawthorne, A.B., Lobb, C.J.: Synchronization and phase locking in two-dimensional arrays of Josephson junctions. *Phys. Rev. B* **53**, 12340, (1996)
- WCM91. Wang, Y.F., Cruz, J.B., Mulligan, J.H.: Two Coding Strategies for Bidirectional Associative Memory. *IEEE Trans. Neu. Net.* **1**(1), 81–92, (1991)
- WCS96. Wiesenfeld, K., Colet, P., Strogatz, S.H.: Synchronization Transitions in a Disordered Josephson Series Array. *Phys. Rev. Lett.* **76**, 404407, (1996)
- WCS98. Wiesenfeld, K., Colet, P., Strogatz, S.H.: Frequency locking in Josephson arrays: Connection with the Kuramoto model. *Phys. Rev. E* **57**, 15631569, (1998)
- WD95. Wang, C.C., Don, H.S.: An analysis of high-capacity discrete exponential bam. *IEEE Trans. Neu. Net.* **6**(2), 492–496, (1995)
- WDH93. Wilmott, P., Dewynne, J., Howinson, S.: *Option Pricing: Mathematical Models and Computation*. Oxford Financial Press, (1993)
- WF49. Wheeler, J.A., Feynman, R.P.: Classical Electrodynamics in Terms of Direct Interparticle Action. *Rev. Mod. Phys.* **21**, 425–433, (1949)
- WFP97. Witt, A., Feudel, U., Pikovsky, A.S.: Birth of Strange Non-chaotic Attractors due to Interior Crisis. *Physica D* **109**, 180–190, (1997)
- WMW01. Weibert, K., Main, J., Wunner, G.: Periodic orbit quantization of chaotic maps by harmonic inversion. *Phys. Lett. A* **292**, 120, (2001)
- WR87. Wonham, W.M., Ramadge, P.J.: On the supremal controllable sublanguage of a given language. *SIAM J. Control Optim.* **25**, 637–659, (1987)
- WS91. Watkin, T.L.H., Sherrington, D.: The parallel dynamics of a dilute symmetric Hebb-rule network. *J. Phys. A: Math. Gen.* **24**, 5427–5433, (1991)

- WS97. Watanabe, S., Swift, J.W.: Stability of periodic solutions in series arrays of Josephson junctions with internal capacitance. *J. Nonlinear Sci.* **7**, 503, (1997)
- WS98. Watts, D.J., Strogatz, S.H.: Collective dynamics of ‘small-world’ networks. *Nature*, **393**, 440-442, (1998)
- WSE00. Waissman, J., Sarrate, R., Escobet, T. et al: Wastewater treatment process supervision by means of a fuzzy automaton model. In *Proc. 2000 IEEE Inter. Symp. Intell. Contr.*, 163-168, Patras, Greece, (2000)
- WSS85. Wolf, A., Swift, J.B., Swinney, H.L., Vastano, J.A.: Determining Lyapunov Exponents from a Time Series. *Physica D*, **16**(3), 285-317, (1985)
- WW83a. Wehner, M.F., Wolfer, W.G.: Numerical evaluation of path-integral solutions to Fokker-Planck equations. I., *Phys. Rev. A* **27**, 2663-2670, (1983)
- WW83b. Wehner, M.F., Wolfer, W.G.: Numerical evaluation of path-integral solutions to Fokker-Planck equations. II. Restricted stochastic processes, *Phys. Rev. A*, **28**, 3003-3011, (1983)
- WZ98. Waelbroeck, H., Zertuche, F.: *Discrete Chaos. J. Phys. A* **32**, 175, (1998)
- Wal81. Walther, H.-O.: Homoclinic solution and chaos in $\dot{x}(t) = f(x(t-1))$. *Nonl. Ana.* **5**, 775-788, (1981)
- Wal89. Walther, H.-O.: Hyperbolic periodic solutions, heteroclinic connections and transversal homoclinic points in autonomous differential delay equations. *Memoirs AMS* **402**, (1989)
- Wal95. Walther, H.-O.: The 2-dimensional attractor of $x'(t) = -\mu x(t) + f(x(t-1))$, *Memoirs AMS* **544**, (1995)
- Wan96. Wang, Z.: A Bidirectional Associative Memory Based on Optimal Linear Associative Memory. *IEEE Trans. Comp.* **45**(10), 1171-1179, (1996)
- Wat90. Watson, L.T.: Globally convergent homotopy algorithms for nonlinear systems of equations. *Nonlinear Dynamics*, **1**, 143-191, (1990)
- Wat99. Watts, D.J.: *Small Worlds*. Princeton University Press, Princeton, (1999)
- Wat99. Watts, D.J.: *Small Worlds: The Dynamics of Networks Between Order and Randomness*. Princeton Univ. Press, Princeton, NY, (1999)
- Wei05. Weisstein, E.W.: MathWorld—A Wolfram Research Web Resource. <http://mathworld.wolfram.com>, (2005)
- Wei90. Weinstein, A.: Affine Poisson structures. *Internat. J. Math.*, **1**, 343-360, (1990)
- Wer89. Werbos, P.J.: Backpropagation and neurocontrol: A review and prospectus. In *IEEE/INNS Int. Joint Conf. Neu. Net.*, Washington, D.C., vol 1, 209-216, (1989)
- Wer90. Werbos, P.: Backpropagation through time: what it does and how to do it. *Proc. IEEE*, **78** (10), (1990)
- Whe61. Wheeler, J.A.: Geometrodynamics and the Problem of Motion. *Rev. Mod. Phys.* **33**, 63-78, (1961)

- Whe62. Wheeler, J.A.: Geometrodynamics. Academic Press, New York, (1962)
- Whi27. Whittaker, E.T.: A Treatise on the Analytical Dynamics of Particles and Rigid Bodies, Cambridge Univ. Press, Cambridge, (1927)
- Whi87. Whitney, D.E.: Historical perspective and state of the art in robot force control. *Int. J. Robot. Res.*, **6**(1), 3–14, (1987)
- Wie61. Wiener, N.: Cybernetics. Wiley, New York, (1961)
- Wig90. Wiggins, S.: Introduction to Applied Dynamical Systems and Chaos. Springer, New York, (1990)
- Wik05. Wikipedia, the free encyclopedia. <http://wikipedia.org>, (2005)
- Wil00. Wilson, D.: Nonlinear Control, Advanced Control Course (Student Version), Karlstad Univ., (2000)
- Wil93. Willmore, T.J.: Riemannian Geometry. Oxford Univ. Press, Oxford, (1993)
- Wil97. Williams, R.M.: Recent progress in Regge calculus. *Nucl. Phys. B* **57**, 73–81, (1997)
- Win01. Winfree, A.T.: The Geometry of Biological Time. Springer, New York, (2001)
- Win67. Winfree, A.T.: Biological rhythms and the behavior of populations of coupled oscillators. *J. Theor. Biol.* **16**, 15, (1967)
- Wol02. Wolfram, S.: A New Kind of Science. Wolfram Media, (2002)
- Wol84. Wolfram, S.: Cellular Automata as Models of Complexity. *Nature*, **311**, 419–424, (1984)
- Woo00. Wooldridge, M.: Reasoning about rational agents. MIT Press, Boston, MA, (2000)
- Woo92. Woodhouse, N.: Geometric Quantization. Clarendon Press, Oxford, (1992)
- Wu82. Wu, F.Y.: The Potts model. *Rev. Mod. Phys.* **54**, 235–268, (1982)
- XH94. Xu, Z., Hauser, J.: Higher order approximate feedback linearization about a manifold. *J. Math. Sys. Est. Con.*, **4**, 451–465, (1994)
- XH95. Xu, Z., Hauser, J.: Higher order approximate feedback linearization about a manifold for multi-input systems. *IEEE Trans. Aut. Con*, AC-40, 833–840, (1995)
- XS05. Xia, Z., Storm, D.R.: The role of calmodulin as a signal integrator for synaptic plasticity. *Nature Rev. Neurosci.* **6**, 267–276, (2005)
- YA01. Yalcin, I., Amemiya, Y.: Nonlinear factor analysis as a statistical method. *Statistical Science*, **16**, 275–294, (2001)
- YAS96. Yorke, J.A., Alligood, K., Sauer, T.: Chaos: An Introduction to Dynamical Systems. Springer, New York, (1996)
- YC98. Yang, T., Chua, L.O.: Control of chaos using sampled-data feedback control. *Int. J. Bif. Chaos* **8**, 2433–2438, (1998)
- YL52. Yang, C.N., Lee, T.D.: Statistical theory of equation of state and phase transitions I: Theory of condensation. *Phys. Rev.* **87**, 404–409, (1952)

- YL96. Yalcinkaya, T., Lai, Y.-C.: Blowout bifurcation route to strange nonchaotic attractors. *Phys. Rev. Lett.* **77**, 50395042, (1996)
- YL97. Yalcinkaya, T., Lai, Y.-C.: Phase Characterization of Chaos. *Phys. Rev. Lett.* **79**, 3885–3888, (1997)
- YML00. Yanchuk, S., Maistrenko, Yu., Lading, B., Mosekilde, E.: Effects of a parameter mismatch on the synchronization of two coupled chaotic oscillators. *Int. J. Bifurcation and Chaos* **10**, 2629–2648, (2000)
- YO05. Yamana, M., Okada, M.: Correlation of Firing in Layered Associative Neural Networks. *J. Phys. Soc. Jpn.* **74**(8), 2260–2264, (2005)
- YSY91. Yanai, H.F., Sawada, Y., Yoshizawa, S.: Dynamics of an Auto-Associative Neural Network Model with Arbitrary Connectivity and Noise in the Threshold. *Network*, **2**, 295–314, (1991)
- YZ99. Yong, J., Zhou, X.: *Stochastic controls. Hamiltonian Systems and HJB Equations*. Springer, New York, (1999)
- Yag87. Yager, R.R.: *Fuzzy Sets and Applications: Selected Papers by L.A. Zadeh*, Wiley, New York, (1987)
- Yeo92. Yeomans, J.M.: *Statistical Mechanics of Phase Transitions*. Oxford Univ. Press, Oxford, (1992)
- ZHC93. Zhuang, X., Huang, Y., Chen, S.S.: Better Learning for Bidirectional Associative Memory. *IEEE Trans. Neu. Net.* **6**(8), 1131–1146, (1993)
- ZK65. Zabusky, N.J., Kruskal, M.D.: Interactions of solitons in a collisionless plasma and the recurrence of initial states. *Phys. Rev. Lett.*, **15**, 240–43, (1965)
- ZS72. Zakharov, V.E., Shabat, A.B.: Exact theory of two-dimensional self-focusing and one-dimensional self-modulation of waves in nonlinear media. *Soviet Physics, JETP* **34**, 62–69, (1972)
- ZM99. Zhang, J., Morris, J.: Recurrent neuro fuzzy models for nonlinear process modeling. *IEEE Trans. Neu. Net.* **10**(2), 313–326, (1999)
- ZTG04. Zumdieck, A., Timme, M., Geisel, T.: Long Chaotic Transients in Complex Networks. *Phys. Rev. Lett.* **93**, 244103, (2004)
- ZWL02. Zheng, F., Wang, Q.-G., Lee, T.H.: Output tracking control of MIMO fuzzy nonlinear systems using variable structure control approach. *IEEE Trans. Fuzzy Syst.* **10**(6), 686–697, (2002)
- ZZ03. Zhu, J.Y., Zhu, H.: Introducing small-world network effects to critical dynamics. *Phys. Rev. E* **67**, 026125, (2003)
- ZZ94. Zurawski, R., Zhou, M.C.: Petri nets and industrial applications - A tutorial. *IEEE Trans. Indust. Elect.* **41**, 567–583, (1994)
- Zad65. Zadeh, L.A.: Fuzzy sets, *Inform. Contr.* **8**, 338–353, (1965)
- Zad78. Zadeh, L.A.: Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, **1**(1), 3–28, (1978)

- Zan02a. Zanette, D.H.: Non-monotonic dependence on disorder in biased diffusion on small-world networks. *Europhys. Lett.* **60**, 945, (2002)
- Zan02b. Zanette, D.H.: Dynamics of rumor propagation on small-world networks. *Phys. Rev. E* **65**, 041908, (2002)
- Zas05. Zaslavsky, G.M.: *Hamiltonian Chaos and Fractional Dynamics*. Oxford Univ. Press, Oxford, (2005)
- Zin93. Zinn-Justin, J.: *Quantum Field Theory and Critical Phenomena*. Oxford Univ. Press, Oxford, (1993)

Index

- 1995, 220
- abstract data type, 41
- accessible automata, 578
- action, 362
- action estimates, 181
- action potential, 5
- action value function, 178
- action–amplitude picture, 532
- activation derivative, 210
- activation dynamics, 211
- acyclic FCM, 555
- Adaline, 186
- Adams–Bashforth–Moulton integrator, 124
- adaptation, 164
- adaptive control, 7
- adaptive filter, 211
- adaptive fuzzy process, 594
- adaptive Kalman filtering, 7
- adaptive path integral, 532, 534, 535
- adaptive resonance theory, 215
- adaptive sensory–motor control, 216
- adaptive signal tracking, 143
- adaptive system, 183
- additive fuzzy system, 258
- adjacency matrix, 554
- adjoint functors, 621
- adjunction, 56
- afferent nerves, 17
- affine fuzzy system, 561
- agent dynamorphism, 155
- agent evolution, 155
- agent theory, 155
- algebra of mental objects, 50
- algorithms, 239
- anatomical MRI, 30
- approximate feedback linearization, 135
- array of N Josephson Junctions, 110
- artificial neural network, 42, 183
- association, 56
- association functor, 52
- association machine, 43
- association–functor, 40
- association–map, 39
- associative array, 39, 41
- associative composition, 595, 613
- associative memory, 8, 39
- associative storage, 39
- associativity of morphisms, 614
- associator, 524
- assumption of symmetric coupling, 147
- asymmetric dilution, 485
- atmospheric convection, 64
- attack function, 216
- attractor, 42
- attractor AM–model, 367
- attractor neural networks, 388
- auditory cortex, 4
- augmented FCM matrix, 559
- auto–overlap, 291, 326
- autocorrelation associative memory, 375
- autonomic nervous system, 13
- axon, 11
- Bäcklund transform, 111

- backpropagation, 191
- base variable, 250
- basin of attraction, 42, 102
- belief–desire–intention agents, 155
- Belousov–Zhabotinski reaction, 85
- Bessel functions, 411, 418
- best known future action, 181
- Betti numbers, 616
- bidirectional associative memory, 296, 375
- bifurcation point, 63
- binary signals, 210
- binding, 41
- biomechanical force–velocity relation, 272
- bipolar neurons, 12
- bipolar signals, 210
- bisimulation, 518
- Bloch, 30
- block entropy, 69
- Boolean derivative, 357
- bootstrapping, 180
- braided monoidal category, 520
- brain dynamorphism, 285
- brain evolution, 285
- brain plasticity, 6, 9
- brain–like control functor, 501
- Brownian motion, 275
- brute–force, 100
- Burgers dynamical system, 117
- C. elegans*, 307
- Cantor set, 231
- capacitively–shunted junction, 448
- capacity dimension, 80, 88
- capacity index, 376
- cascaded recurrent ANN, 207
- categories and functors, 43
- category of neurons, 47
- category of things, 51
- cellular dynamorphism, 229
- cellular evolution, 228
- cellular robotics, 594
- center manifold, 328
- central biodynamic adjunction, 9
- central nervous system, 12
- cerebellar cortex, 3
- cerebellum, 28
- cerebral cortex, 3
- cerebral hemispheres, 28
- chain coupling, 272
- channel multiplexing, 298
- chaos control, 86
- chaos theory, 64
- chaotic attractor, 102
- chaotic behavior, 64
- chaotic system, 67
- Chapman–Kolmogorov equation, 288
- characteristic Lyapunov exponents, 87
- Chervonenkis SVM theory, 218
- Christoffel symbols, 120
- chromosome, 243
- Chua’s circuit, 73, 103
- Chua–Matsumoto circuit, 85
- classification, 174, 217
- co–active neuro–fuzzy inference system, 201
- cognition, 37
- cognitive agent, 38
- cognitive architecture, 37
- cognitive information processing, 215
- Cohen–Grossberg activation equations, 216
- Cohen–Grossberg theorem, 216
- coherence laws, 524
- coherent quantum oscillations, 108
- collective link, 44
- combined FCM, 556
- combined FCM matrix, 559
- combining noisy sensor outputs, 123
- competitive learning, 338
- complement, 250
- complete synchronization, 347
- complex networks, 360
- complex system/process, 61
- complex–valued ANNs, 208
- complex–valued order parameter, 353
- composition of functors, 56
- computer networking device, 40
- concept space, 174
- concurrent machine, 513
- conditional Lyapunov exponents, 354
- conformal z –map, 79
- conjugate gradient method, 194
- connection matrix, 554
- connectionist, 187
- connectionist approach, 183
- connectivity matrix, 307

- content-addressable memory, 367
- content-addressed memory, 39
- context space, 207
- continual time systems, 122
- continuity equation, 361
- continuous Hopfield network, 290
- continuous phase transition, 270
- continuous-time, 61
- continuum limit equation, 361
- control, 61, 167
- control affine system, 129
- control parameter, 270
- control parameters, 268, 269
- control vector-fields, 128
- controllable, 576, 581
- controlled trajectory, 129
- controlled Van der Pol oscillator, 133
- convex optimization problem, 203
- Cooper pair box, 108
- Cooper pairs, 107
- Corinna Cortes, 220
- corrector, 125
- correlation function, 275
- correlation-based learning, 376
- correlation-type associative memory, 459
- cost, 151
- covariant force functor, 54, 120, 501, 546
- covariant force law, 53, 271, 501, 546
- covariant formalism on smooth manifolds, 118
- Crick-Koch binding hypothesis, 9
- crisp, 248
- critical current, 109
- critical slowing down, 269, 270, 295
- cross-overlap, 292, 327
- cross-talk noise, 459
- crossing, 244
- crossover, 240
- crosstalk noise, 378
- Curie-Weiss law, 403
- curve, 249
- cycle, 71
- cyclic FCM, 555
- damping rate, 102
- Darwinian evolution principle, 237
- data and goals, 174
- debugging, 167
- decision ladder, 172
- decision-maker, 162
- deductive reasoning agents, 157
- defuzzification, 257, 263, 513
- degree of anisotropy, 36
- degree of order, 269
- Delaunay triangulation, 569
- deliberation reasoning, 168
- delta-rule, 190
- dendrites, 11
- DENFIS system, 266
- depolarization, 18
- design, 167
- deterministic automaton, 576
- deterministic chaos, 65
- diagnosis, 167
- dictionary, 41
- diencephalon, 28
- differential Hebbian law, 216
- diffusion anisotropy, 33
- diffusion coefficient, 31
- diffusion magnetic resonance imaging, 31
- diffusion MRI, 30
- diffusion tensor, 30
- diffusion-weighted images, 33
- directed cycle, 555
- directed homotopy, 516
- directed paths, 516
- directed percolation, 360
- Dirichlet tessellation, 286
- discontinuous phase transition, 271
- discrete event system, 574
- discrete time systems, 122
- discriminant functions, 183
- disorder $\Rightarrow$ order, 271
- disorder-averaged generating functional, 462
- disorder-averaged partition function, 472
- distributive fusion, 595
- dot product, 219
- drift vector-field, 128
- dual system, 124
- Duffing-Van der Pol equation, 116
- dynamic programming, 41, 176, 179
- dynamical evolution, 61
- dynamical recognizers, 207

- dynamical system, 62, 556
- dynamics of n -functors, 120
- dynamorphism, 62
- echo attenuation, 34
- echoplanar imaging, 35
- effective path measure, 462
- efferent functional stimulation, 347
- efficacy, 309
- EGreedy policy, 182
- Einstein's summation convention, 1
- Either-Or, 248
- elastic net algorithm, 286
- elite, 240
- elitism strategy, 245
- embedding field equation, 345
- embedding methods, 195
- empty, 577
- encoding scheme, 175
- enteric nervous system, 13
- entropic function, 402
- entropy, 268
- environment, 162
- equilibrium point, 105
- equilibrium statistical mechanics, 388, 400
- Erdős-Rényi random network, 150
- ergodic system, 391
- ergodicity breaking, 389
- Ericksonian hypnotherapy, 43
- error bound, 218
- Error function, 467
- escape rate, 73
- Euler characteristic, 546
- evolutionary algorithm, 238
- evolutionary algorithms, 239
- evolutionary computation, 203
- evolutionary programming, 237
- evolutionary robotic problem, 572
- evolutive system, 47
- excitatory, 22
- existence and uniqueness theorems for ODEs, 62
- expert systems, 39, 167
- exploiting, 182
- exploration/exploitation problem, 181
- exploring, 182
- extended fuzzy transition function, 578
- extended Hamilton oscillator, 279
- extended Kalman filter, 127, 194
- external configuration manifold, 118
- F, 624
- FCM digraph, 559
- feature-space, 340
- feedback, 556
- feedback fuzzy systems, 264
- feedback linearization, 130
- feedforward neural network, 185
- ferromagnet, 270
- Feynman diagram, 116
- Feynman path integral, 532
- Feynman-Vernon formalism, 322
- Fick law, 145
- finite map, 41
- finite-step sequence processing, 459
- first-order-search methods, 194
- fitness, 240, 245
- FitzHugh-Nagumo neuron, 333
- fixed-point, 42, 556
- Floquet exponents, 455
- Floquet multiplier, 94
- Floquet stability analysis, 93
- flow, 62
- flow pattern, 75
- fluctuating force, 275
- fluctuation-dissipation theorem, 276, 438
- fluctuation-dissipation theorems, 439
- Fokker-Planck equation, 295, 318
- foliation, 141
- force equation, 272
- force functor, 53
- force-field psychodynamics, 532
- forced nonlinear oscillators, 75
- forgetful memories, 292
- formalism of jet bundles, 76
- Fourier transformations, 466
- fractal dimension, 43, 229
- fractals, 79
- fractional dimension, 79
- free energy, 400
- free energy potential, 270
- frequency noise, 305
- frequency-phase modulation, 298
- Frobenius theorem, 141
- frontal lobe, 3
- full coupling, 272

- fully recurrent networks, 204
- function, 52
- function approximators, 254
- function with a finite domain, 41
- functional approximators, 183
- functional modeling machinery, 52
- functional MRI, 30
- functional programming language, 42
- functor, 52
- fuzzification, 255
- fuzzy associative memory, 259
- fuzzy automaton, 578, 580
- fuzzy brain dynamorphism, 547
- fuzzy brain evolution, 547
- fuzzy cognitive map, 552
- fuzzy controllability, 575
- fuzzy diffusion parameter, 254
- fuzzy expert system, 246
- fuzzy IF-THEN rules, 260
- fuzzy inference engine, 255
- fuzzy language, 578, 579
- fuzzy logic, 246
- fuzzy logic controller, 265
- fuzzy membership function, 247
- fuzzy mutual entropy, 253, 254
- fuzzy parameter space, 254
- fuzzy patches, 254
- fuzzy Petri nets, 575
- fuzzy sets, 167
- fuzzy supervisor, 581
- fuzzy transition function, 578
- fuzzy variable, 250
- fuzzy wave equation, 254
- fuzzy-system dynamorphism, 246
- fuzzy-system evolution, 246

- G, 624
- game, 206
- game theory, 206
- game-playing strategies, 237
- Gamma function, 104
- gauge-invariant phase difference, 449
- Gaussian approximation, 468
- Gaussian measure, 468
- Gaussian random field, 464
- Gaussian saddlepoint approximation, 116
- general natural system, 46
- generalized feedforward network, 198
- generalized Gaussian, 211
- generalized Hebbian rule, 352
- generalized policy iteration, 180
- generalized solution, 85
- generalized synchronization, 347
- generating functional, 362
- genetic algorithm, 176, 203
- genetic algorithms, 193, 237, 239
- genetic control, 203
- genetic operators, 240
- genetic programming, 237, 241
- genotype, 241, 243
- geometrodynamical functor, 534
- giant strongly connected-component, 222
- Ginzburg-Landau equation, 117
- Glauber dynamics, 293, 308
- glia, 12
- global associative machinery, 51
- global fuzzy dynamical model, 561
- global solutions, 240
- globally-coupled Kuramoto model, 446
- goodness, 240
- Grünwald-Letnikov definition, 104
- grade of membership, 249
- grade of membership., 248
- gradient descent method, 193
- gradient information, 187
- gradient of the performance surface, 187
- Gray code, 242
- greedy action, 182
- Green's function, 273
- grey matter, 3
- group action, 610
- growth inhibition effects, 230

- Hénon map, 99
- Hamilton oscillator, 278
- Hamiltonian chaos, 100
- Hamiltonian energy function, 300
- Hamming hypercube, 291
- Hartman-Grobman theorem, 294
- HDA-homotopy, 519
- Hebb rules, 7
- Hebbian innovation, 293
- Hebbian learning, 337, 595
- Hebbian rule, 303, 308
- Hebbian synapses, 471
- Hebbian theory, 22

- Hebbian weights, 368
- Hecht–Nielsen counterpropagation network, 213
- heterogeneous pattern association model, 375
- heuristic IF–THEN rules, 160
- heuristic search, 174
- hidden pattern, 556
- hierarchical associative memory organization, 43
- hierarchical self-programming, 489
- high-resolution agent model, 594
- higher-dimensional automata, 516
- higher-order contact, 55
- higher-order tangency, 55
- Hilbert space, 203, 219
- Hindmarsh–Rose thalamic neuron, 334
- Hodgkin–Huxley, 19
- Hodgkin–Huxley equation, 335
- Hodgkin–Huxley neuron model, 334
- homeostasis, 26
- homological algebra, 616
- homology group, 616
- homotopical analysis of concurrency, 514
- homotopy methods, 195
- homotopy operators, 136
- Hopf bifurcation, 102
- Hopf-like bifurcation, 74
- Hopfield discrete-time network, 281
- Hopfield model, 7, 297, 375
- Hopfield overlap, 423
- Hopfield overlaps, 405
- Hopfield synaptic matrix, 292
- human biodynamics, 53
- human brain, 3
- human heart beat and respiration, 347
- human-like locomotor system, 271
- hybrid dynamical system of variable structure, 83
- hybrid systems, 85
- hyperbolic system, 63
- hyperbolic tangent threshold activation functions, 186
- hyperplane, 217
- hyperpolarization, 18
- hysteresis effect, 269, 271
- in modulus, 304
- in the sense of Filippov, 85
- independent and identical distribution, 379
- independent component analysis networks, 201
- index file, 41
- inductive limit, 44
- inference, 257, 260
- infimal controllable fuzzy superlanguage, 591
- infinite set, 248
- infinite-dimensional neural network, 532
- infinite-step sequence processing, 459
- information, 86
- inheritance, 239
- inhibitory, 22
- input vector-fields, 128
- instability, 75
- instability sequence, 269
- instanton vacua, 116
- instruction, 595
- integrate-and-fire neuron, 332
- integrate-and-fire-or-burst neuron, 332
- integration, 62
- intention, 168
- internal configuration manifold, 118
- internal models, 207
- interpretation, 167
- intersection, 250, 577
- inversion, 244
- involution closure, 140
- irregular and unpredictable, 65
- Ising Hamiltonian, 289, 291
- Ising-spin, 231
- Ising-spin Hamiltonian, 400
- Ising-spin Hopfield network, 213
- Ising-spin model, 309
- Ising-spin neuron, 460
- Ising-spin-glass model, 297
- Iterated Prisoner's Dilemma, 206
- iterative maps, 63
- jerk function, 100
- jet bundle, 55
- Jordan and Elman networks, 199
- Josephson current, 107
- Josephson current–phase relation, 108
- Josephson effect, 107

- Josephson interferometer, 107
- Josephson junction, 106
- Josephson tunnelling, 107
- Josephson voltage–phase relation, 109
- Josephson–junction quantum computer, 108
- Kalman filter, 123
- Kalman filtering problem, 125
- Kalman regulator, 123
- Kalman–Bucy filter, 125
- Kaplan–Yorke dimension, 88, 564
- Karhunen–Loeve covariance matrix, 212, 292
- kernel, 219
- kernel trick, 217, 219
- kick equation, 273
- Klein–Gordon equation, 113
- knowledge base, 157
- knowledge representation, 174
- Kohonen algorithm, 286
- Kohonen continuous self organizing map, 214
- Kohonen self-organizing map, 199
- Kolmogorov–Sinai, 89
- Kolmogorov–Sinai entropy, 69, 86, 89, 355, 564
- Kortevég–de Vries equation, 529
- Korteweg–de Vries equation, 111
- Kramers–Moyal expansion, 420
- Kuramoto model, 152, 351, 361, 363
- Kuramoto order parameter, 455
- Kuramoto–Sivashinsky equation, 117
- ladder, 447
- lag synchronization, 347
- Lagrange multipliers, 220
- Lagrangian density, 537
- Landau, 268
- Langevin rate equation, 275
- largest eigen–diffusivity, 37
- largest Lyapunov exponent, 86
- Lattice Green’s function, 466
- Lauterbur, 30
- leaf space, 141
- learn by trial and error, 176
- learning algorithm, 175
- learning dynamics, 211, 212
- learning operations, 174
- learning rate, 188
- learning rate scheduling, 188
- least means square algorithm, 188
- Levenberg–Marquardt algorithm, 194
- Levi–Civita connection, 120
- Lewinian force–field theory, 533
- Lie derivative, 83
- Lie product, 142
- Lie–derivative neuro–classifier, 343
- Lie–Poisson neuro–classifier, 344
- limit cycle, 42, 76, 460, 556
- linear classification, 217
- linear classifier, 219
- linear controllability, 134
- linear homotopy ODE, 84
- linear homotopy segment, 84
- linear quadratic regulator, 123
- linguistic variable, 249, 251
- Liouville measure, 268
- Liouville theorem, 268
- loading parameter, 301
- loading rate, 376
- local fuzzy dynamical model, 561
- local solutions, 240
- locally–coupled Kuramoto model, 447, 456
- logistic map, 91
- long–range connections, 456
- long–range order, 108
- long–term memory, 211
- lookup table, 41
- Lorenz attractor, 83
- Lorenz equations, 99
- Lorenz system, 64
- Lyapunov criterion, 144
- Lyapunov dimension, 88
- Lyapunov exponent, 66, 67, 148
- Lyapunov exponents, 42
- Lyapunov function, 213, 423
- Lyapunov stability, 63
- Lyapunov time, 66
- MAC address, 40
- machine learning, 143
- magnetic resonance imaging, 30
- main direction of diffusivities , 36
- Mamdani, 260
- Mamdani fuzzy controller, 263
- Mandelbrot and Julia sets, 79

- Mansfield, 30
- map sink, 42
- mapping, 41
- margin, 217
- Markov chain, 69, 391
- Markov decision process, 177
- Markov process, 69
- Markov stochastic process, 535
- master stability function, 148
- match-based learning, 215
- matrix cost function, 193
- maximum correlation, 311
- maximum-entropy, 210
- maximum-margin hyperplane, 217
- maxmin rule, 578
- McCulloch-Pitts neurons, 186
- McCumber parameter, 452
- mean diffusivity, 36
- mean kinetic energy, 276
- mean square error, 187
- mean-field, 278
- mean-field theory approximation, 368
- mean-field theory dynamics, 368
- means-ends reasoning, 168
- measurement equation, 125
- medulla oblongata, 28
- membership function, 249
- memoization, 41
- memorized, 308
- memory evolvable system, 47
- memory recall, 174
- memory term, 102
- mental force law, 546
- microtubules, 311
- midbrain, 28
- millimetric image resolution, 32
- minimizing the error, 187
- minimum-time reward functions, 178
- modeling, 61
- modification, 56
- modular feedforward networks, 199
- momentum learning, 192
- monitoring, 167
- monoid, 520
- monoidal category, 520
- Monte-Carlo method, 179
- Moore-Penrose pseudoinverse, 292, 327
- Morris-Lecar neuron, 334
- motor cortex, 4
- motor nerve fibers, 13
- multi-agent systems, 156, 594
- multi-kick equation, 274
- multi-spiral strange attractor, 74
- multilayer perceptron, 185, 196
- multipolar neurons, 12
- mutation, 239, 240, 244
- mutual overlap, 292
- Nash equilibrium, 206
- natural geometrical structures, 519
- natural selection, 239
- natural transformation, 56
- Navier-Stokes equations, 234
- nearest neighbor coupling, 272
- Necker cube, 280
- network dynamorphism, 183
- network evolution, 183
- network topology, 175
- networks of dynamical elements, 147
- neural communication, 5
- neural network, 7, 183
- neural path integral, 496
- neural state-space, 296
- neuro-muscular adjunction, 57
- neurons, 10
- Newton-Raphson method, 195
- Newtonian mechanics, 62
- Newtonian method, 193
- next chosen action, 181
- Noetherian ring, 612
- noise filtering, 174
- noise-free limit, 396
- non-autonomous 2D continuous systems, 75
- non-equilibrium phase transitions, 295
- non-equilibrium statistical mechanics, 388
- nonfuzzy, 248
- nonlinear classification, 217
- nonlinear control system, 128
- nonlinear controllability criterion, 140
- nonlinear dynamics and chaos, 61
- nonlinear function approximation, 254
- nonlinear MIMO-systems, 128
- nonlinear optics, 298
- nonlinear Schrödinger (NLS) equation, 114
- nonlinear Schrödinger equation, 528

- normalized, 249
- object-oriented programming, 156
- occasional proportional feedback, 564
- occipital lobe, 3
- ODEs, 63
- Oja-Hebb learning rule, 211
- olfactory cortex, 4
- operator, 52
- optimal estimator problem, 124
- optimal hyperplane, 217
- optimal performance, 595
- optimal policy, 596
- optimal value function, 179
- optimization, 174, 218
- orbit, 62, 70
- order parameter, 268, 270
- order parameter equation, 270, 294, 378, 407
- order parameters, 388
- order-parameter, 363
- Ornstein-Uhlenbeck process, 432
- oscillator neural networks, 351
- overdamped, 448
- overdamped ladder, 456, 458
- overdamped limit, 109
- overlap, 291, 326
- parallel composition, 580
- parameter update law, 144
- parametric neural network, 298
- parasympathetic nervous system, 13
- parietal lobe, 3
- partition function, 400
- path-integral, 362
- path-integral formalism, 469
- path-integral method, 459
- pattern, 42, 44
- pattern matching process, 215
- pattern recognition, 174
- pattern-recognition, 189
- Pavlov, 22
- Pavlov's conditional-reflex learning, 22
- perceptron, 189
- perfect environment model, 179
- performance parameter, 369
- performance surface, 187
- period-doubling bifurcation, 91
- periodic orbit theory, 73
- periodic phase synchronization, 348
- periodic solutions, 75
- periodicity, 464
- peripheral nerve fibers, 13
- peripheral nervous system, 12
- phase coherence, 108
- phase difference, 109, 348
- phase noise, 305
- phase point, 75
- phase portrait, 75
- phase synchronization, 347
- phase transition, 270
- phase transition of first order, 270
- phase transition of second order, 270
- phase-flow, 62
- phase-transition effects, 269
- phase-transition theory, 268
- phenotype, 241, 243
- Philip Hall basis, 142
- physically-controllable systems of nonlinear oscillators, 447
- pinball game, 66
- planning, 167
- Poincaré section, 93
- Poincaré section, 72
- Poincaré-Bendixson theorem, 75
- Poisson process, 275
- policy, 176
- pons, 28
- population's fitness, 239
- positive leading Lyapunov exponent, 88
- postsynaptic potential, 289
- Potts model, 297
- Potts vectors, 299
- Potts-glass neural network, 298
- predictability time, 88
- prediction, 61
- prediction/forecasting, 167, 174
- predictor, 124
- primary term, 250
- principal component analysis networks, 201
- principle of incommensurability of frequencies, 303
- Prisoners' Dilemma, 206
- pristine, 456
- probability distribution function, 451
- process equation, 125
- product, 580

- production systems, 39
- production-rule agents, 157
- progress graph, 515
- projective limit, 44
- pruning, 70
- psychological continuum, 249
- Purcell, 30
- pure delayed reward functions, 178
- Pyragas control, 93
- Qlearning, 181
- quadratic cost function, 124
- quadratic I&F neuron, 333
- quadratic programming, 218
- quantum brain, 532
- quantum computers, 107
- quantum-mechanical wave function, 107
- qubits, 107
- Rössler, 100
- radial basis function, 219
- radial basis function network, 201
- random force, 362
- random homogeneous network, 147
- random optimization, 241
- random search, 239
- rank condition, 138
- rate of error growth, 86
- Rayleigh-Bénard convection, 64, 85
- re-polarization, 18
- reachable sets, 129
- recognize-act cycle, 168
- recombination, 240
- recurrence time, 113
- recurrent ANN, 207
- recurrent fuzzy voronoi (RFV) model, 566
- recurrent neural networks, 388
- recursively hierarchical structure, 3
- reflectance pattern, 214
- reflex, 25
- regional energy minimization, 37
- regression, 217
- regular lattice, 147
- regularized, 219
- Reidemeister moves, 523
- reinforcement learning, 176, 595, 596
- relative degree, 82, 133
- repeller, 72
- replica symmetric, 459
- resistive loading, 109
- resistively-shunted junction, 448
- resonance, 108
- resonate-and-fire neuron, 333
- return map, 72
- reward-signal, 176
- Riemann curvature tensor, 545
- Riemann-Liouville definition, 104
- Riemannian geometry, 54
- Rosenblatt, 189
- roulette, 245
- roulette wheel algorithm, 238
- roulette wheel selection, 240
- route to chaos, 102
- rule-importance weight, 595
- rules fire, 261
- saddle-point integration, 403
- saltatory conduction, 20
- Sarsa, 180
- scale-free network, 147
- schedule, 516
- Schrödinger equation, 114
- Schwann cells, 12
- search, 187
- select, 240
- selection, 240
- selection method, 245
- self-organizing feature maps, 286
- semi-parametric classifiers, 183
- sensation, 162
- sensitive dependence on initial conditions, 65
- sensor fusion in hybrid systems, 128
- sequence of period doublings, 102
- sequential (threshold) dynamics, 289
- sequential machine, 513
- set, 248
- set of high mountains, 248
- set of things, objects, points, 51
- set operation, 250
- sets, 248
- Sherrington-Kirkpatrick model, 363, 489
- short-term memory, 211
- signal velocity, 211
- signal-to-noise analysis, 459

- simplicial set, 525
- simulated annealing, 193
- simulated neural network, 183
- sine-Gordon (SG) equation, 113
- Sine-Gordon equation, 529
- singleton, 249
- situation, 162
- situation awareness, 162
- slope parameter, 186
- small world, 448
- small-time locally controllable, 129
- small-world geometry, 152
- small-world network, 147
- small-world networks, 152, 360, 447
- SMO algorithm, 219
- social hierarchy, 152
- social network, 152
- social organization analysis, 172
- socio-economic model, 557
- soft margin method, 220
- SoftMax action selection, 182
- solitons, 110
- solution, 84
- somatic nervous system, 13
- somatosensory cortex, 4
- source category, 52
- space, 219
- sparse coding, 373
- spatio-temporal pattern, 42
- spatiotemporal networks, 216
- spinal nerves, 18
- stability, 75
- Standard Additive Model, 258
- state, 162
- state manifold, 128
- state value function, 178
- state-space explosion problem, 519
- static backpropagation, 186
- statistical neurodynamics, 380, 459
- steepest descent method, 188
- step function, 222
- step size, 188
- stochastic learning process, 288
- stochastic-gradient order parameter equations, 294
- strange attractor, 42, 64, 65, 232
- structural instability, 268
- M machinery, 52
- structural stability, 63
- sub-net configuration, 41
- subsumption architecture, 159
- superconductivity, 107
- supervised learning, 217
- supervised network, 185
- support, 249, 577
- support vector machine, 202
- support vector machines, 217
- support vector regression, 220
- support vectors, 217
- supremal controllable fuzzy sublanguage, 585
- survival probability, 73
- symbolic dynamics, 70
- symmetric dilution, 485
- symmetry breaking instability, 269, 295
- sympathetic nervous system, 13
- symplectic geometry, 54
- synaptic junctions, 6
- synchronization dynamics, 147
- synchronization in chaotic oscillators, 347
- synchronization of coupled nonlinear oscillators, 446
- synchronization of coupled oscillators, 147
- synchronized population coding, 366
- synergetics, 267, 280, 294
- synergetics dynamorphism, 267
- synergetics evolution, 267
- synfire chain, 384
- synfiring, 385
- synnaptic interaction matrix, 460
- system dynamorphism, 122
- system evolution, 122
- system input-output relation, 254
- system parameters, 272
- system with uncertain dynamics, 123
- Takagi-Sugeno (TS) fuzzy model, 560
- tangle, 521
- tangle diagram, 522
- target category, 52
- taste, 249
- temperature value, 182
- temporal associative memories, 292, 557
- temporal dynamical systems, 212
- temporal lobe, 3
- temporary phase-locking, 347

- tensor-field, 62
- term set, 250
- the parametric neural network, 303
- the saddle-node bifurcation, 328
- thermal equilibrium, 268
- thermodynamic limit, 462
- thermodynamics, 234
- theta-neuron, 328, 333
- time-difference method, 179
- time-lagged recurrent networks, 203
- time-phase plot, 76
- Toda lattice, 114
- Toeplitz matrix, 465
- M approach, 37
- topological entropy, 67
- topological operads, 525
- totalistic Boolean CA, 356
- tournament method, 238
- tournament selection, 245
- training data, 217
- trajectory, 62, 70, 75
- trans-derivational search, 43
- traveling wave solution, 112
- tricategories, 524
- turbulence, 63
- uncertain environment, 61
- uncertainty, 167
- uncontrollable, 581
- underdamped junction, 448
- underdamped ladder array, 456
- union, 250, 577
- unipolar neurons, 12
- universal approximation theorem, 196
- universe of discourse, 248, 249
- unstable periodic orbits, 73
- value function, 178
- Van der Pol oscillator, 82
- Vapnik, 218
- M lism, 298
- vector-field, 62
- velocity equation, 272
- vertebrate brain, 27
- virtual communication links, 345
- visual cortex, 4
- Volterra-Lotka ensemble dynamics, 296
- Voronoi diagram, 569
- Voronoi tessellation, 286
- water vapor, 269
- Watts-Strogatz model, 307
- weakly-connected neural network, 9
- weakly-connected neural networks, 294
- weighted category, 47
- white matter, 3
- Wigner function, 321
- Winfree-type phase models, 446
- Winner-relaxing Kohonen algorithm, 288
- work domain analysis, 171
- work organization analysis, 172
- Yang-Mills relation, 116
- young, 249
- young people, 248