KATHOLIEKE UNIVERSITEIT LEUVEN
FACULTEIT TOEGEPASTE WETENSCHAPPEN
DEPARTEMENT ELEKTROTECHNIEK
Kasteelpark Arenberg 10, 3001 Leuven (Heverlee)

CELLULAR NEURAL NETWORKS, MULTI-SCROLL CHAOS AND SYNCHRONIZATION: THEORY, APPLICATIONS AND IMPLEMENTATIONS

Jury:
Prof. H. Van Brussel, voorzitter
Prof. J. Vandewalle, promotor
Prof. J. A. K. Suykens, promotor
Prof. D. Aeyels (Ghent University)
Prof. A. Barbé
Prof. D. Roose
Prof. T. Roska (Hungarian Academy of Sciences)
Prof. M. Steyaert

Proefschrift voorgedragen tot
het behalen van het doctoraat
in de toegepaste wetenschappen

door

Müştak E. Yalçın

U.D.C. 681.3*F11

June 2004

© Katholieke Universiteit Leuven – Faculteit Toegepaste Wetenschappen
Arenbergkasteel, B-3001 Heverlee (Belgium)

Alle rechten voorbehouden. Niets uit deze uitgave mag vermenigvuldigd en/of openbaar gemaakt worden door middel van druk, fotocopie, microfilm, elektronisch of op welke andere wijze ook zonder voorafgaande schriftelijke toestemming van de uitgever.

All rights reserved. No part of the publication may be reproduced in any form by print, photoprint, microfilm or any other means without written permission from the publisher.

D/2004/7515/56

ISBN 90-5682-519-4

Foreword

At the end of my doctoral work it is a pleasure to thank those who have helped me along the way.

First of all, I would like to express my thanks to Prof. Joos Vandewalle who has made this opportunity possible. I would also like to thank him for the many opportunities I have received to broaden my knowledge and discuss my work with other researchers.

I would like to thank Prof. Johan Suykens who is my promoter, a colleague and a friend that I have had the privilege to enjoy. This thesis is the result of a close collaboration and stimulating discussions we had together. Furthermore, he sacrificed many hours correcting preprints of my articles and earlier draft of this thesis.

I also would like to thank Prof. André Barbé, Prof. Dirk Aeyels and Prof. Dirk Roose for reviewing this manuscript.

I am very grateful to Prof. Tamás Roska from the Hungarian Academy of Sciences for accepting to be a member of the jury. His leading research on cellular neural/nonlinear networks served as a source of inspiration for my research. I would also like to thank him for interesting discussions whenever we met.

Prof. Hendrik Van Brussel is gratefully acknowledged for chairing the jury.

I would like to acknowledge the collaboration with (in alphabetical order): Prof. Paolo Arena, Adriano Basile, Prof. Luigi Fortuna, Assist. Prof. Serdar Özoğuz and Samuel Xavier de Souza. I have learned many things from them and working with them was a great pleasure.

A part of this work has been realized within the framework of the EU-funded research project DICTAM (IST 1999-19007). I would like to thank Prof. Angel Rodríguez-Vázquez who coordinated the project and Dr. Servando Espejo-Meana, Dr. Ákos Zárandy, Dr. Luigi Occhipinti and Dr. Pierre van den Bulck for stimulating discussions during the project.

I would like to thank all my colleagues for the excellent working atmosphere. Special thanks to Oscar Barrero, Steven Bex, Tijl De Bie, Jos De Brabanter, Bart Hamers, Lukas, Kristiaan Pelckmans, Jan Schier, Maarten Van den Nest and Fritz von Haeseler. I greatly appreciate the efforts of Bart Motmans, Pela Noe, Ilse Pardon and Ida Tassens for their help with the administrative matters.

The financial support of the Research Council Scholarships programme of the Katholieke Universiteit Leuven and the Interuniversity Attraction Poles programmes which is managed by the Belgian Federal Science Policy Office (PPS Science Policy) formerly called Federal Office for Scientific, Technical and Cultural Affairs (O.S.T.C.), are greatly appreciated.

The adventure of this doctoral work began with my studies at the Division of Circuits and Systems of İstanbul Technical University, Electrical and Electronic Engineering Faculty. I would like to thank Prof. Cüneyt Güzeliş who inspired to me to become a researcher, for his support and shared ideas with me. I also would like to thank Prof. Acar Savacı without his encouragement this work would not have been possible. I do not find a word to say how rich and rewarding my life has been because of them. I thank Prof. Fuat Anday for allowing to be on a leave from the İstanbul Technical University.

I would like to thank Bilge Bayrakcı, F. Kağan Gürkaynak, Ebru Şahin, Kanat Çamlıbel, Eralp and Deniz Tezcan who have always been there for the moral support and a good old chat.

I specially want to thank my mother, İffet, my father, Yıldırım, and my sister, Ersan. Over the years they have continued to support and encourage me in my studies and my doctoral work. *Anneme, babama ve ablama candan teşekkürler.*

I want to thank my wife Berna for her love and happiness she is bringing in my life. One of the best experiences that we have had during our study was the birth of our daughter Setenay. She brings some noises and chaos, as well as lots of happiness to our life.

Müştak E. Yalçın

Leuven, June 2004

Abstract

In this thesis, we investigate applications that aim at exploiting complex dynamics of nonlinear systems. We study Cellular Neural/Nonlinear Networks (CNNs) which are architectural models for obtaining programmable networks. CNNs consist of an array of nonlinear dynamical cells and the instructions are represented by interconnection weights. An important related class of nonlinear systems investigated in this thesis are Lur'e systems. We show that Lur'e systems such as generalized Chua's circuits are able to exhibit multiple equilibria, periodic behaviour, bifurcations and chaos. We discuss realizations and present experimental confirmations of 3−, 5− and 6−scroll attractors from generalized Chua's circuits. Furthermore, we present a new family of scroll grid attractors. We show that for this family the generated scrolls can be located in any state variable direction. Consequently $n \times m$- and $n \times m \times k$-grid $(n, m, k \in \{1, 2, 3, \dots\})$ scroll attractors are constructed. In a second part of the thesis we present a systematic overview of synchronization methods for Lur'e systems. Experimental confirmations of synchronization schemes for autonomous and non-autonomous cases with double and 5-scroll attractors are given together with circuit realizations. A time delay-synchronization scheme is designed to prevent the scheme against propagation delay. Finally, in a third part of the thesis, complex dynamics of nonlinear systems (individual cells as well as collective behaviour) are employed towards applications. New applications are made in the areas of optimization, cryptography, image and video processing. In order to overcome the problem of unwanted local minima of gradient networks, we study the use of chaotic signals for global optimization via coupled chaotic annealing. This admits the study of cooperative behaviour within a coupled local minimizers and CNN framework. Furthermore, applications of chaos as a source for random number generators in cryptography are studied. We present a random bit generator based on a double scroll attractor which successfully passes all statistical tests of the FIPS-140-1 and Diehard test suite. Finally, we introduce a fragile watermarking method for CNNs in image and video authentication problems for multimedia applications. All results are demonstrated on CNN-Universal chip prototypes.

Notation

Mathematical Notation

A^T	transpose of the matrix A
$\mathbb{R}^n$	set of n-dimensional real vectors
$\mathbb{R}^{n\times m}$	set of real matrices of dimension $n \times m$
C^p	$p-$times continuously differentiable
∇f	gradient of a real-valued function f
$[a, b]$	the closed interval between two real numbers
$\lvert\cdot\rvert$	absolute value of a real number
$\lVert x\rVert_2$, $x \in \mathbb{R}^n$	2-norm of vector x, $\lVert x\rVert_2 = (\sum_{i=1}^{n} \lvert x_i\rvert^2)^{1/2}$
$\lVert A\rVert_2$, $A \in \mathbb{R}^{n\times m}$	2-norm of matrix A, $\lVert A\rVert_2 = (\lambda_{\max}(A^T A))^{1/2}$
$\lambda_i(A)$	i-th eigenvalue of A
$\sigma_{max}(A)$	maximal singular value of A
$\oplus$	logic exclusive-or
$\vee$	logic or
diag(v)	diagonal matrix having the elements of vector v on its diagonal
Im x	imaginary part of the complex number x
$0_{M\times N}$	$M \times N$ zero matrix
I_N	identity matrix of size $N \times N$

Acronyms and Abbreviations

1-D	One Dimensional
2-D	Two Dimensional
ACE	Analogic Cellular Engine
AMC	Analogic Machine Code
CA	Cellular Automaton
CAn	Chaotic Annealing

CCA	Coupled Chaotic Annealing
CCPS	CNN Chip Prototyping System
CDMA	Code-Division Multiple Access
CFOA	Current Feedback Operational Amplifier
CLM	Coupled Local Minimizer
CMOS	Complementary Metal-Oxide-Silicon
CNN	Cellular Neural/Nonlinear Network
CNN-UM	CNN Universal Machine
CSK	Chaotic Shift Keying
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DWT	Discrete Wavelet Transform
FIPS	Federal Information Processing Standards
GN	Gradient Network
HRNG	Hybrid Random Number Generator
I/O	Input/Output
IC	Integrated Circuit
LFSR	Linear Feedback Shift Register
LLM	Local Logic Memory
LSB	Least Signifiant Bit
MNS-CNN	Multiple Neighborhood Size CNN
MOSFET	Metal Oxide Semiconductor Field Effect Transistor
MOSIS	Metal Oxide Semiconductor Implementation Service
NIST	National Institute of Standards and Technology
NUP-CNN	Non-Uniform Processor CNN
ODE	Ordinary Differential Equation
Opamp	Operational Amplifier
PDE	Partial Differential Equation
PLC	Predicted Limit Cycle
PPG	Pseudorandom Pattern Generator
PWL	Piecewise Linear
RNG	Random Number Generator
SC-CNN	State Controlled CNN
StISC	Spatial-temporal Instruction Set Computer
TRNG	True Random Number Generator
TSP	Traveling Salesman Problem
VCCS	Voltage-Controlled Voltage Source
VCVS	Voltage-Controlled Current Source
VF	Voltage Follower
VLSI	Very Large Scale Integration
VW2D	Variable-Watermark Two-Dimensional

Contents

Foreword i

Abstract iii

Notation v

Contents vii

1 Introduction 1

1.1 Nonlinear systems in engineering 1

1.2 This thesis 2

1.3 Chapter by chapter overview 4

1.4 Main contributions 10

2 Cellular Neural/Nonlinear Networks 13

2.1 CNN 13

2.1.1 1-D CNNs 14

2.1.2 2-D CNNs 16

2.2 CNN models 19

2.2.1 Chua-Yang CNN model 19

2.2.2 State controlled CNN (SC-CNN) model 20

2.2.3 Full-range CNN model 21

2.2.4 Reaction-diffusion CNN model 21

2.2.5 Generalized CNN models 23

2.2.5.1 A generalized CNN model: nonlinear and delay CNNs 23

2.2.5.2 Another generalized CNN model 24

2.2.5.3 A generalized CNN based on Chua's circuit . 25

2.3 CNN universal machine: a visual microprocessor 25

2.4 New research aspects for CNNs 28

2.4.1 Wave computing algorithm 28

2.4.2 Coupled local minimizers 30

2.5 Conclusion 34

3 Multi-scroll Chaotic and Hyperchaotic Attractors 35

3.1 Chua's circuit 36

3.1.1 Chaos in Chua's circuit 37

3.1.1.1 The Shil'nikov method 37

3.1.1.2 Harmonic balance method 42

3.1.2 Realization of Chua's circuit 46

3.2 Generalized Chua's circuit 48

3.2.1 Realization of n-scroll attractors from generalized Chua's circuits 52

3.2.1.1 Realization of 3- and 5-scroll attractors 54

3.2.1.2 On the realization of n-scroll attractors 59

3.2.1.3 Alternative realizations of n-scroll attractors . 60

3.3 Families of scroll grid attractors 63
3.3.1 A new family of n-scroll attractors 65
3.3.2 2-D grid scroll attractors 66
3.3.3 3-D grid scroll attractors 70
3.3.4 Circuit realizations 73
3.3.5 Alternative realizations 82
3.4 Multi-scroll hyperchaotic attractors 82
3.4.1 Hyperchaotic n-scroll attractors 83
3.4.2 n-Scroll hypercube attractors 84
3.5 Scroll maps from n-scroll attractors 86
3.5.1 1-Scroll and 2-scroll maps 88
3.5.2 Circuit realization of a 1-scroll map 90
3.6 Lur'e representations 92
3.7 Conclusion 96

4 Synchronization of Chaotic Lur'e Systems 99

4.1 Synchronization 100
4.2 Master-slave synchronization: autonomous case 103
4.2.1 Full static state error feedback 103
4.2.2 Dynamic output feedback 105
4.3 Robust synchronization 107
4.3.1 Full static state error feedback 107
4.3.2 Dynamic output feedback 110
4.4 Time-delay synchronization scheme 112
4.4.1 Error system of time-delay synchronization scheme . . . 113

4.4.2 Delay-dependent synchronization criterion 115
4.4.3 Full static state feedback together with time delay . . . 118
4.5 Nonlinear H_∞ synchronization: non-autonomous case 119
4.5.1 Full static state error feedback 119
4.5.2 Dynamic output feedback 123
4.6 Robust nonlinear H_∞ synchronization 124
4.6.1 Full static state error feedback 125
4.6.2 Dynamic output feedback 127
4.7 Impulsive synchronization 128
4.7.1 State feedback case 129
4.7.2 Measurement feedback case 130
4.8 Controller design 131
4.8.1 Master-slave synchronization 132
4.8.2 Robust synchronization 133
4.8.3 Synchronization with time-delay 133
4.8.4 Nonlinear H_∞ synchronization 134
4.8.5 Robust nonlinear H_∞ synchronization 134
4.9 Examples 135
4.9.1 Master-slave synchronization of two unidirectionally coupled Chua's circuits 135
4.9.2 Experimental confirmation of time-delay synchronization scheme 138
4.9.3 Experimental confirmation of nonlinear H_∞ synchronization for Chua's circuit and 5-scroll attractors 140
4.10 Conclusion 146

5 Engineering Applications 149

5.1 Chaos in communications 149
5.1.1 Modulation and demodulation based on chaos 150
5.1.1.1 Static modulation methods 151
5.1.1.2 Dynamic modulation methods 151
5.1.2 The role of new synchronization schemes 152
5.2 Chaotic systems in optimization 154
5.2.1 Chaotic annealing 155
5.2.2 Coupled chaotic annealing 156
5.2.3 Illustrative examples 158
5.3 Random number generators and cryptography 160
5.3.1 TRNG from a double scroll attractor 162
5.3.1.1 Bit generation 164
5.3.1.2 De-skewing and setting the threshold 168
5.3.1.3 Statistical tests 169
5.3.2 Binary pseudorandom pattern generation on CNN-UM . 172
5.3.2.1 PPGs from 1-D rules 174
5.3.2.2 PPGs from 2-D rules 176
5.3.2.3 Experimental results on the CNN-UM chip . . 176
5.3.2.4 A general comparison 178
5.4 Image/Video authentication on CNN-UM 180
5.4.1 Fragile watermarking 182
5.4.2 Fragile watermarking on the CNN-UM for image authentication 182
5.4.3 Fragile watermarking on the CNN-UM for video authentication 184
5.4.4 Experimental results on the CNN-UM chip 184

5.5 CNN template tuning 186
5.5.1 Erroneous chip behavior and template robustness 187
5.5.2 Experiments 188
5.6 Conclusions 190

6 General conclusions and future work **193**

Bibliography **197**

Chapter 1

Introduction

In this Introduction, we first give a motivation for the use of nonlinear systems in engineering. In Section 1.2, we sketch the broad picture of this thesis based on the motivation in Section 1.1. In Section 1.3, we outline the content of each chapter. In Section 1.4, we state the main contributions and new results of this thesis.

1.1 Nonlinear systems in engineering

What would be the reason to study nonlinear systems instead of linear systems which have been used extensively in engineering to model and control the behavior of dynamical systems for an engineer? Our answer to this question reminds us to the answer of Sastry [223] to the question "Why do we need a nonlinear theory?". The reason is that dynamics of linear systems are not rich enough to describe many commonly observed phenomena. A few examples of such phenomena are

a. Multiple equilibria or multiple operating points,

b. Periodic behaviour of state variables or limit cycles,

c. Bifurcation,

d. Chaos.

Of course, from a scientific point of view these phenomena are very interesting. Therefore nonlinear dynamics has become an interdisciplinary field of science including physics, mathematics and many other sciences. From the engineering point of view, although a curiosity to observe and understand these phenomena on a system fascinates engineers, the main objective for them is exploiting these phenomena in possible applications. A Hopfield network [114] which is used as an associative memory, transiently chaotic neural networks [35], chaos communication [1] and active wave computing [208] are some of the well-know examples which use multiple equilibria, bifurcation, synchronization and complex dynamical behavior in their proposed applications, respectively. For an associative memory, equilibria of the Hopfield network correspond to several patterns which one wants to store. Transiently chaotic neural networks use bifurcation phenomena such that the network can explore the search space for the global optimum for a given optimization problem. In chaos communications, while the chaotic signal is used as a carrier for transmitting information, synchronization is exploited to recover the information from the carrier. Active wave computing techniques which exploit spatial-temporal waves, are used in image processing applications.

When the above-mentioned phenomena are considered as tools to solve engineering problems there is a need for a platform which might handle the computational complexity of these phenomena. In fact, cellular neural/nonlinear network (CNN) [48] is such a framework. It is followed by the CNN universal machine (CNN-UM) [212] which is the first algorithmically programmable analog array computer. Hence CNN-UM has created a platform to exploit the above-mentioned phenomenas. Today we witness physical implementations of the CNN-UMs [161, 162, 23] which have a profound impact in the engineering community such that it brings these phenomena from mathematical abstraction into the core of electrical engineering.

Figure 1.1 shows two different perspectives in order to deal with engineering applications. One which is the classical approach, is based on digital signal processing, the other which is motivated by dynamics of nonlinear systems, addresses the same problems. *Analogic* (from the contraction of *analog* and *logic*) computers [212] exploit both perspectives with their dual computing capability.

1.2 This thesis

In this thesis, we investigate applications that aim at exploiting the dynamics of nonlinear systems (see Figure 1.1). We start with introducing CNNs which will

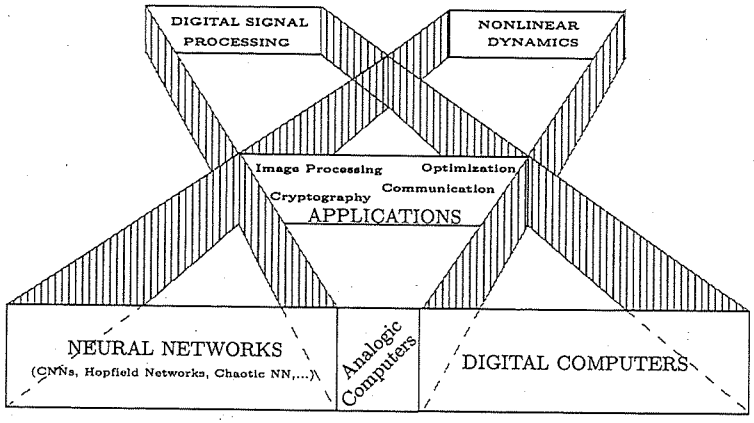


Figure 1.1: Two different perspectives can be taken here to solve engineering problems. While the classical computers basically handle the problems based on digital signal processing techniques, a new perspective from nonlinear dynamic studies addresses the same problem via neural networks. Analogic computers based on CNN-UMs are the platforms which are at the intersection of these two perspectives.

Application	Section	Chapter 2	Chapter 3	Chapter 4
Communications	Section 5.1		×	×
Optimization	Section 5.2	×	×	×
True Random Number Generator	Section 5.3.1		×	
Pseudorandom Pattern Generator	Section 5.3.2	×		
Image/Video authentication	Section 5.4	×		

Table 1.1: The applications and the Chapters which are exploited for the applications (× indicates which Chapters are involved).

be the architectural model to obtain programmable networks. These networks are composed of nonlinear dynamic units. An important class of nonlinear systems studied in this thesis are Lur'e systems, which can be represented as a linear system interconnected by feedback to a nonlinearity that satisfies a sector condition. Chapter 3 shows that a Lur'e system such as the Chua's circuit and generalized Chua's circuit are able to exhibit multiple equilibria, periodic variation, bifurcation and complex dynamical behavior. Synchronization issues which are addressed in Chapter 4 are linked to the absolute stability problem of Lur'e systems which has been well studied in classical nonlinear control theory. In Chapter 5 these phenomena are individually and collectively employed and in synergy with the applications. Table 1.1 shows the applications and the related chapters. A flowchart for this thesis is given in Figure 1.2.

1.3 Chapter by chapter overview

Chapter 2. Cellular neural/nonlinear networks

This Chapter is devoted to Cellular Neural/Nonlinear Networks which are locally coupled nonlinear dynamical systems. We start presenting the definition of the CNN acronym which is independent from its interpretation as Cellular Neural Networks or Cellular Nonlinear Networks. We also present its mathematical definition by four specifications. The technical and special terms which are used in the CNNs research are explained. Then we introduce the well-known CNN models. We also review two generalized CNN models. While the result of one of these indicates that most of the known neural network architectures are covered by the generalized CNN model, the other leads to two new models, i.e. delay type and nonlinear CNNs. Then the CNN universal machine

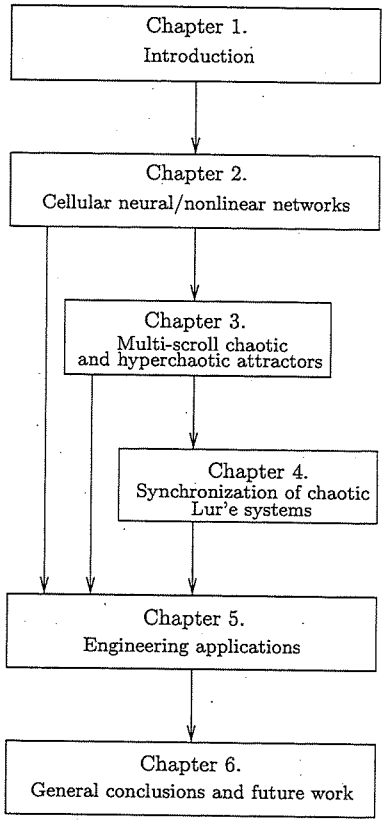


Figure 1.2: Flowchart for this thesis. Chapter 5 collects the applications which individually and synergistically exploit the previous chapters. The applications and the chapters which are used in these applications are shown in Table 1.1.

Date	Contribution	Reference
1986	Double scroll family	Chua *et al.* [43]
1991	n-double scroll attractors	Suykens *et al.*[245, 246]
1996	2-double scroll attractor implementation	Arena *et al.* [6]
1997	n-scroll attractors	Suykens *et al.* [244]
1999	6-scroll attractor implementation	Yalçın *et al.* [301]
2000	3- and 5-scroll attractors implementation	Yalçın *et al.* [302]
2001	n-scroll attractors via sine function	Tang *et al.* [262]
	Families of scroll grid attractors	Yalçın *et al.* [312]
	1-D grid scroll attractors	
	2-D grid scroll attractors	
	3-D grid scroll attractors	
2002	10-scroll attractor implementation	Zhong *et al.* [325]
2003	Prototype chip for a 3-scroll attractor	Fujiwara *et al.* [86]
	1-D and 2-D grid scroll attractors from second-order linear system with hysteresis.	Han *et al.* [104]

Table 1.2: Chronology of multi-scroll chaotic attractors, milestones and contributing authors.

(CNN-UM) is introduced together with recent technological developments. We aim at reflecting the results of the VLSI hardware realizations of CNN-UMs (and also CNNs) without going into details. Next the background materials are surveyed and new research aspects are given. Wave computing which is a new research aspect, is driven by the VLSI implementations of CNN-UMs. We present the basic concepts and meaning of new terms of wave computing. Another research aspect considered in this Chapter is Coupled Local Minimizers (CLM). CLMs are presented here under the CNN interpretation and linked to wave computing. The CNN framework is further explored in the rest of this thesis.

Chapter 3. Multi-scroll chaotic and hyperchaotic attractors

We give an extensive overview of multi-scroll chaotic and hyperchaotic attractors. First we present multi-scroll chaotic attractors and our outline for this Chapter is illustrated on Table 1.2 which chronologically shows authors and their contributions to the research on multi-scroll attractors. Historically, the double scroll attractor from Chua's circuit is probably the first multi-scroll attractor. Therefore we start multi-scroll chaotic attractors with Chua's circuit.

We collect the theoretical and the heuristic approaches which prove the existence of the chaotic behavior on Chua's circuit. We provide a short overview for circuit realizations of Chua's circuit, and give a circuit realization for Chua's circuit. This realization methodology allows to adjust independently the slopes and break points of the nonlinearity and is exploited to realize a generalized Chua's circuit. Then we discuss realizations and present experimental confirmations of 1-, 3- 5- and 6-scroll attractors. We also introduce a scaling of the PWL nonlinearity of the generalized Chua's circuit such that n-scroll attractors for large n values become feasible for implementations. We collect alternative realizations of n-scroll attractors based on Chua's circuit. Then a new family of scroll grid attractors is presented. These families are classified into three classes called 1-D, 2-D and 3-D grid scroll attractors depending on the location of the equilibrium points in state space. The scrolls generated from 1-D, 2-D and 3-D grid scroll attractors are located around the equilibrium points on a line, on a plane or in 3-D, respectively. Due to the generalization of the nonlinear characteristics, it is possible to increase the number of scrolls in all state variable directions. A number of strange attractors from the scroll grid attractor families are presented. They have been experimentally verified using current feedback opamps. We collect the recent research directions on the family of scroll grid attractors and their realizations. Then we focus on hyperchaotic attractors and introduce multi-scroll hyperchaotic attractors. While hyperchaotic n-scroll attractors are obtained from a generalized model like the generalized Chua's circuit, n-scroll hypercubes attractors are observed on a 1-D CNN consisting of n-scroll cells with weak unidirectional or diffusive coupling between the cells. Then we introduce scroll maps which generated imitated behavior of a two-side Poincaré map of the n-scroll attractors. Finally we present Lur'e representations of the multi-scroll chaotic and hyperchaotic circuits which are studied in this Chapter. Hence, many results concerning stability and synchronization are applicable to it.

Chapter 4. Synchronization of chaotic Lur'e systems

In this Chapter we give a systematic overview of methods for master-slave synchronization in the case of Lur'e systems. First we introduce the coupling configurations which help to distinguish between synchronization schemes. Then we focus on the master-slave coupling configuration which implies that the master system evolves freely and drives the dynamics of the slave system. We basically divide the synchronization schemes into autonomous and non-autonomous schemes. For autonomous schemes the link is made between synchronization and absolute stability theory of Lur'e systems. First, the master-slave synchronization schemes and its synchronization criteria are given for both the full static state feedback and dynamic output feedback cases (when the full

state vector can not be measured). The proofs of the synchronization criteria are also given. A second scheme for the autonomous case is the robust synchronization scheme which deals with two non-identical Lur'e systems taking into account the influence of parameter mismatch between the master and the slave systems. The synchronization criteria and their proofs for the robust synchronization are given for both full static state feedback and dynamic output feedback cases. Then we deal with propagation delay in the master-slave synchronization scheme. A criterion for master-slave synchronization of Lur'e systems is presented when a time-delay exists in the master and slave systems. Synchronization which is either time-dependent or time-independent is studied based on a Lyapunov-Krasovskii function for global asymptotic stability of the error system. A delay-dependent synchronization criterion is given based on a new Lyapunov function. Then we introduce non-autonomous schemes which are basically motivated by the research on exploring the applications of chaotic systems in communications. For non-autonomous schemes the link is made between synchronization and nonlinear H_∞ theory. Using a full state error or dynamic output feedback mechanism and formulating the synchronization problem with the standard plant framework of modern control theory, it is shown how the message signal can be recovered for a continuous time reference input (or message signal) which drives the master system. Synchronization criteria and their proofs for the nonlinear H_∞ synchronization are given for both full static state feedback and dynamic output feedback cases. We also study the influence of parameter mismatch between the master and slave systems with respect to the method of nonlinear H_∞ synchronization. As a last synchronization scheme, impulsive synchronization is discussed for state and dynamic measurement feedback. It has to be pointed out that all of the given criteria are global and are sufficient conditions. Furthermore the conditions are expressed in terms of matrix inequalities which lead to optimization based design of the controllers for achieving synchronization. Hence all of the given synchronization criteria correspond to related optimization problems. This may help the reader who wishes to skip theoretical derivations and design a controller for his/her applications at hand. We provide an algorithmic procedure to employ a synchronization scheme (Figure 1.3) which also shows how Chapters 3 and 4 provide solutions for each block. At the end of this Chapter, we illustrate a number of examples which experimentally confirm the theoretical results. We use the circuits that are designed in the previous Chapter in order to implement a master-slave and a nonlinear H_∞ synchronization scheme.

Chapter 5. Engineering applications

In this Chapter, we face the question of how the results in the previous Chapters can be used in applications. Table 1.1 shows the applications, which are

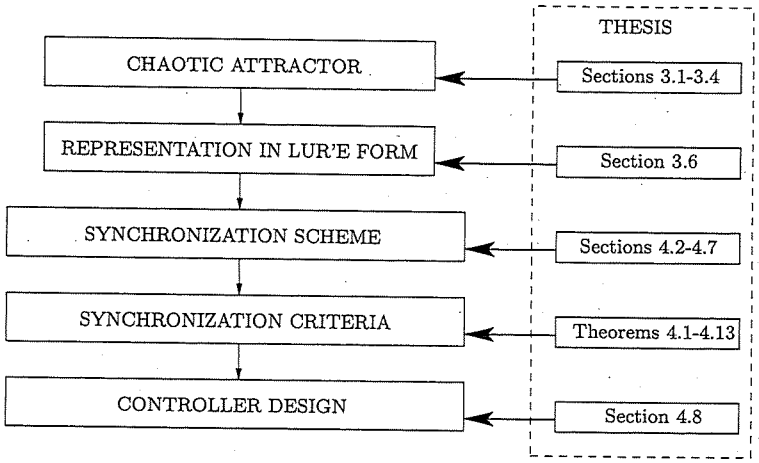


Figure 1.3: An algorithmic procedure to implement the synchronization schemes studied in this thesis.

considered in this thesis, and the Chapters which are used for those applications. We first present chaos communications. Although chaos has potential applications in several functional blocks of a communication system, we focus on applications on the modulation and demodulation blocks. First we provide the reader with an overview on schemes which have exploited chaos for these blocks. Then we discuss and highlight the role of synchronization schemes from Chapter 4 for these blocks. Our discussions take into account the recent results of the studies on chaos communications. A new application for chaos is found in optimization networks. First, we give an overview on the methods which have been presented in the literature. Then we introduce two new methods which are chaotic annealing and coupled chaotic annealing. Chaotic annealing is similar to the use of noise in continuous simulated annealing where the noise is replaced by chaos. Coupled chaotic annealing integrates chaotic annealing methods within the coupled local minimizers method. This method is interpreted in the CNN framework such that two 1-D CNNs work cooperatively towards finding improved local minima. The method can also be considered as a wave computing algorithm which is controlled by the state synchronization. A next application of chaotic signals in engineering is its use as a source for random number generators. Here we present a random bit generator which uses a double scroll attractor from a simple circuit model. The proposed TRBG is subjected to statistical tests using the well-known tests suites FIPS-140-1

and Diehard in cryptography. The proposed TRBG successfully passes all the statistical tests of the FIPS-140-1 test suite and a strong and more complete test suite of Diehard. We also apply the Cellular Automata for pseudorandom pattern generation and extend the related algorithms under the framework of the CNNs. The introduced algorithms are tested on a CNN chip. The last application exploits the CNN-UMs for a watermarking-based image and video authentication and the pseudorandom pattern generators are used as a watermark generator. On-chip experimental results are reported, confirming the suitability of CNN-UMs to successfully act in real-time watermarking. Although VLSI implementations of the CNN-UMs (and the CNNs) derive many new directions in CNNs research, we focus on a side effect of their VLSI implementations at the end. Instead of seeking the reasons of erroneous behavior on the highly complex CNN-UMs chips, we directly approach the problem by generating tuned templates.

1.4 Main contributions

In this section we state the main contributions and new results of this thesis. The following results are new:

- *Implementation of n-scroll attractors from generalized Chua's circuit:*
  We present an experimental confirmation of 3- and 5-scroll attractors from a generalized Chua's circuit. The piecewise linear characteristic of a generalized Chua's circuit for generating n-scroll attractor is synthesized by a nonlinear VCVS which is realized by operational amplifiers in such a way that this realization allows that the slopes and break points of the nonlinearity can be independently adjusted. Also we discuss scaling properties of the nonlinearity in a generalized Chua's circuit. After the scaling, n-scroll attractors for larger n values become feasible for implementation. This contribution can be found in Sections 3.2.1.1 and 3.2.1.2. References are [301] and [302].

- *Families of scroll grid attractors:*
  A new family of scroll grid attractors is presented. These families are classified into three classes called 1-D, 2-D and 3-D grid scroll attractors depending on location of the equilibrium points related to the location of the equilibrium points in state space. The scrolls generated from 1-D, 2-D and 3-D grid scroll attractors are located around the equilibrium points on a line, on a plane or in 3-D, respectively. Due to the generalization of the nonlinear characteristics, it is possible to increase the number of scrolls in all state variable directions. They have been experimentally

verified using current feedback opamps. Also Lur'e representations are given for the scroll grid attractor families. This contribution can be found in Section 3.3. References are [313], [311], [310], [312] and [252].

- *Hyperchaotic n-scroll attractors and scroll maps:*
  Hyperchaotic n-scroll attractors which are generated from a generalized Matsumoto-Chua-Kobayashi circuit, are obtained. Furthermore, n-scroll maps are constructed through the Poincaré cross section in the n-scroll attractor. Explicit equations for the 1- and 2-scroll maps are given. Also a 1-scroll map has been verified with its simple circuit realization. This contribution can be found in Sections 3.4.1 and 3.5. References are [303], [314] and [252].

- *Time-delay synchronization scheme:*
  Time-delay effects on the master-slave synchronization scheme are investigated. Sufficient conditions for master-slave synchronization of Lur'e systems are presented for a known time-delay in the master and slave systems. A delay-dependent synchronization criterion is given based upon a new Lyapunov-Krasovskii function. The derived criterion is a sufficient condition for global asymptotic stability of the error system, expressed by means of a matrix inequality. The feedback matrix follows from solving a nonlinear optimization problem. The method is illustrated for the synchronization of Chua's circuits, 5-scroll attractors and hyperchaotic attractors. This contribution can be found in Sections 4.4.1, 4.4.2 and 4.9.2. References are [304], [305] and [253].

- *Experimental confirmations of synchronization schemes:*
  An experimental confirmation of a master-slave synchronization scheme with double scroll attractors is given with a circuit realization. Furthermore experimental confirmations of a nonlinear H_∞ synchronization scheme with double and 5-scroll attractors are presented also with a circuit realization. For both schemes the controllers are designed based on the synchronization criteria for Lur'e systems. This contribution can be found in Sections 4.9.1 and 4.9.3. References are [299], [300] and [253].

- *Chaotic annealing and coupled chaotic annealing:*
  Methods of chaotic simulated annealing within the context of coupled local minimizers are formulated. Interpreted within the cellular nonlinear networks context, coupled local minimizers consider local optimization algorithms as cells with local connections between the cells. As a result, information exchange is taking place between the minimizers. Instead of taking local optimization methods as individual cells, we explore the use of chaotic signals as additional driving force as in continuous simulated annealing where deterministic chaos now plays the role of noise. On a

number of examples, improved results are obtained by coupled chaotic annealing. In general, the coupling of the minimizers also leads to a variance reduction on the optimal cost function values simulated for many different runs. This contribution can be found in Sections 5.2.1 and 5.2.2. Reference is [259].

- *True random bit generation from a double scroll attractor:*
  A novel true random bit generator (TRBG) based on a double scroll attractor is proposed. The double scroll attractor is obtained from a simple model which is qualitatively similar to Chua's Circuit. In order to face the challenge of using the proposed TRBG in cryptography, the proposed TRBG is subjected to statistical tests which are the well-known FIPS-140-1 and Diehard test suits in the area of cryptography. The proposed TRBG successfully passes all these tests and can be implemented in integrated circuits. This contribution can be found in Section 5.3.1. References are [306] and [307].

- *Image and Video authentication on the CNN-UM:*
  A new approach to the fragile watermarking technique is introduced. This problem is particularly interesting in the field of modern multimedia applications, when image and video authentication are required. The approach exploits the Cellular Automata suitability to work as Pseudo-random Pattern Generators and extends the related algorithms under the framework of the cellular nonlinear networks. The result is a novel way to perform watermarking generation in real time, using the presently available CNN-UM chip prototypes. Both the CNN algorithms for fragile watermarking as well as on-chip experimental results are reported, confirming the suitability of CNNs to successfully act as real-time watermarking generators. The availability of CNN-based visual microprocessors allows to have powerful algorithms to watermark in real time images or videos for efficient smart camera applications. This contribution can be found in Sections 5.3.2, 5.4.2 and 5.4.3. References are [308], [9] and [309].

Chapter 2

Cellular Neural/Nonlinear Networks

In this Chapter, we provide the reader with background material on Cellular Neural/Nonlinear Networks (CNNs) together with their recent technological developments. Furthermore we introduce two new research directions in CNNs which are active wave computing and coupled local minimizers. This Chapter is organized as follows. After a broad and formal definition of CNN in Section 2.1, the well-known models and generalizations of the CNN are presented. Then, in Section 2.3 we review the CNN Universal Machine (CNN-UM) with the recent developments in its dedicated VLSI hardware realizations and computational infrastructures. In Section 2.4 two new research directions for CNNs are introduced.

2.1 CNN

CNN was first introduced as an acronym for *Cellular Neural Network* by Chua and Yang [48] in 1988. It is an information or signal processing system composed of a large number of simple analog processing elements, called cells, which are locally interconnected and which cooperate to perform parallel processing in order to solve a given computational task. The key concept that distinguishes a CNN from other neural networks [327] is that the interconnections among cells are local. This is indeed a great advantage which makes CNN structures tailor-made for monolithic implementation in currently available planar

technologies [161, 162, 214]. Since its invention many applications of CNNs have been presented in literature [216, 294, 231, 232, 274, 215, 8], especially for image-processing tasks [46].

In 1993, Chua and Roska [45] also interpreted CNN as *Cellular Nonlinear Network*, employed for generating chaotic and hyperchaotic signals [124, 257, 198], static and dynamic patterns, autowaves, spiral waves, spatial-temporal chaos [39, 40]. From an engineering point of view, CNN characterizes a network consisting of locally coupled nonlinear dynamical systems to generate behaviour or to process information. Here instead of struggling to distinguish between the interpretations, we will define and then study CNN acronym itself. In addition to this reason, in this thesis CNNs are used for processing (Section 5.4) and producing (Section 3.4.2) information. Furthermore, these two purposes are combined in coupled local minimizers and coupled chaotic annealing process (Section 2.4.2 and 5.2.2, respectively).

Definition 2.1 (CNN) [40]. *A CNN is any spatial arrangement of locally-coupled cells, where each cell is a dynamical system which has an input, an output and a state evolving according to some prescribed dynamical laws.*

The resulting net is defined mathematically by four specifications [41, 171]:

- Cell dynamics,
- Synaptic law,
- Boundary conditions and
- Initial conditions.

A 1-D (Figure 2.1(a)) and a 2-D (Figure 2.1(b)) CNN architecture is a spatial arrangement of locally-coupled of a row of N cells and an $N \times M$ array of cells, respectively. For a continuous-time CNN, cells consist of time-invariant circuit elements (see Figure 2.3) and the cells can be any dynamical system. Although the definition 2.1 includes discrete-time CNNs [107], we only consider continuous-time CNNs throughout this thesis and we also assume here that the cells are spatially invariant. A 3-D CNN is built up by cascading 2-D CNN layers.

2.1.1 1-D CNNs

A 1-D CNN cell will be denoted by $\mathfrak{C}_i$ where $i \in \{1, 2, \ldots, N\}$. Each cell in the 1-D CNN architecture is a dynamical system and coupled locally only to

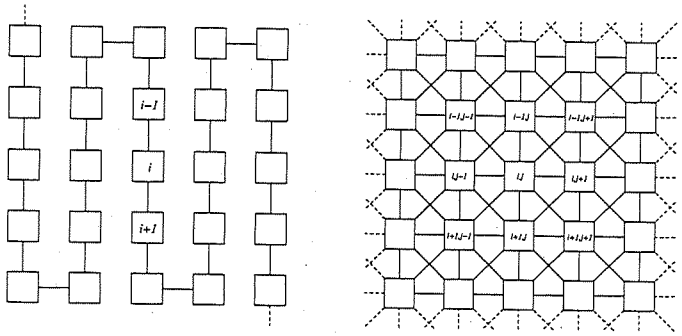


Figure 2.1: Cells (each cell is represented by a square) which are spatially arranged in (a) a 1-D and (b) a 2-D CNN architectures.

those neighboring cells which lie inside a prescribed *sphere of influence* $\mathcal{S}_i(r)$ of radius r (r is a positive integer), where [40]

$$\mathcal{S}_i(r) = \{\mathfrak{C}_k : \max(|k-i|) \leq r,\ 1 \leq k \leq N, k \neq i\}. \tag{2.1}$$

The cell dynamics are defined by the state equations

$$\begin{aligned} \dot{x}_i &= g(x_i, z_i, u_i(t), I_i^s) \\ y_i &= f(x_i) \end{aligned} \tag{2.2}$$

with state vector $x_i \in \mathbb{R}^{m_x}$, output vector $y_i \in \mathbb{R}^{m_y}$, threshold $z_i \in \mathbb{R}^{m_z}$ and input vector $u_i \in \mathbb{R}^{m_u}$ of the i–th cell $\mathfrak{C}_i$. $I_i^s \in \mathbb{R}^{m_I}$ is a synaptic law of the cell $\mathfrak{C}_i$. $f : \mathbb{R}^{m_x} \to \mathbb{R}^{m_y}$ is nonlinear output function and $g : \mathbb{R}^{m_x} \times \mathbb{R}^{m_z} \times \mathbb{R}^{m_u} \times \mathbb{R}^{m_I} \to \mathbb{R}^{m_x}$. The threshold z_i is usually assumed to be a constant scalar (denoted as z).

The synaptic law defines the coupling between the considered cell $\mathfrak{C}_i$ and all the cells within the prescribed sphere of influence $\mathcal{S}_i$

$$\begin{aligned} I_i^s = \sum_{k \in \mathcal{S}_i} \Big\{ & \hat{A}_{k-i} x_k + A_{k-i} y_k + B_{k-i} u_k + A_{k-i}^\tau y_k(t-\tau) \\ & + B_{k-i}^\tau u_k(t-\tau) + A_{k-i}^n(x_i, x_k) + B_{k-i}^n(u_i, u_k) \Big\}. \end{aligned} \tag{2.3}$$

The symbol $k \in \mathcal{S}_i$ denotes the index of all cells $\mathfrak{C}_k$ lying inside $\mathcal{S}_i$. The synaptic law is uniquely specified by synaptic matrices $\hat{A}_{\hat{k}} \in \mathbb{R}^{m_I \times m_x}$, $A_{\hat{k}} \in \mathbb{R}^{m_I \times m_y}$,

$B_{\hat{k}} \in \mathbb{R}^{m_I \times m_u}$, $A_{\hat{k}}^\tau \in \mathbb{R}^{m_I \times m_y}$, $B_{\hat{k}}^\tau \in \mathbb{R}^{m_I \times m_u}$ and functions $A_{\hat{k}}^n : \mathbb{R}^{m_x} \times \mathbb{R}^{m_x} \to \mathbb{R}^{m_I}$ and $B_{\hat{k}}^n : \mathbb{R}^{m_u} \times \mathbb{R}^{m_u} \to \mathbb{R}^{m_I}$ where $\hat{k} \in \{-r, -r+1, \cdots, r-1, r\} \setminus \{0\}$. These matrices together with $\hat{A}_0$, A_0, B_0, *etc.* are usually listed as entries of the matrices, referred to as *state template* $\hat{A}$ [5], *feedback template* A [48], *feedforward template* (or *control template*) B [48], *delay-type feedback template* A^τ [211], *delay-type feedforward template* B^τ [211], *nonlinear feedback template* $A^n(x_i, x_k)$ [211] and *nonlinear feedforward template* $B^n(u_i, u_k)$ [211]. The superscripts τ and n of a template indicate that the template is a delay-type and nonlinear type template, respectively. The synaptic matrices $A_{\hat{k}}$ are often collected in the feedback template

$$A = \begin{bmatrix} A_{-r} & A_{-r+1} & \cdots & A_0 & \cdots & A_{r-1} & A_r \end{bmatrix}. \tag{2.4}$$

The templates A, B and the threshold z are called the *cloning template.*

Every CNN cell has the same sphere of influence, except if it is located on an edge of the finite-sized array. Boundary conditions are defining the neighborhood of these cells. It has been shown that the dynamical behavior of an important group of CNNs strongly depends on these boundary conditions [263]. The most common boundary conditions are:

1. *Fixed (Dirichlet) Boundary Condition:*

$$x_0 = E_1, \tag{2.5}$$
$$x_{N+1} = E_2 \tag{2.6}$$

   where $E_1, E_2 \in \mathbb{R}$

2. *Zero-Flux (Neumann) Boundary Condition:*

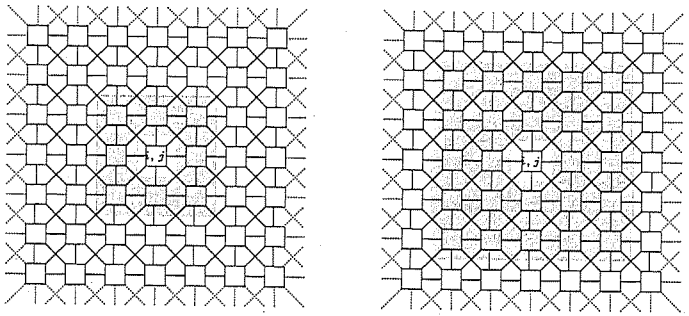
$$x_0 = x_1 \tag{2.7}$$
$$x_{N+1} = x_N \tag{2.8}$$

3. *Periodic (Toroidal) Boundary Condition:*

$$x_0 = x_N \tag{2.9}$$
$$x_{N+1} = x_1. \tag{2.10}$$

2.1.2 2-D CNNs

A 2-D CNN cell will be denoted by $\mathfrak{C}_{i,j}$ where $i \in \{1, 2, \ldots, N\}$ and $j \in \{1, 2, \ldots, M\}$ and each cell is coupled locally only to those neighboring cells



(a) (b)

Figure 2.2: A CNN cell $\mathfrak{C}_{i,j}$ with (a) a 3×3 sphere of influence (a neighborhood of radius 1) and (b) a 5×5 sphere of influence (a neighborhood of radius 2). The cells $\mathfrak{C}_{k,l}$ which are in the sphere of influences $\mathcal{S}_{i,j}$ are indicated by the gray area.

which lie inside a prescribed sphere of influence $\mathcal{S}_{i,j}(r)$ of radius r, where [40]

$$\mathcal{S}_{i,j}(r) = \{\mathfrak{C}_{k,l} : \max(|k-i|, |l-k|) \leq r,\ 1 \leq k \leq N, \\ 1 \leq l \leq M, k \neq i, l \neq j\}. \tag{2.11}$$

A 3×3 (corresponding to $r=1$) and 5×5 (corresponding to $r=2$) sphere of influence are shown in Figures 2.2(a) and 2.2(b), respectively. In these cases, $\mathfrak{C}_{i,j}$ are coupled to the 8 and 24 nearest neighboring cells $\mathfrak{C}_{k,l}$ which are indicated by the gray area in Figures 2.2(a) and 2.2(b), respectively. The cell dynamics of the 2-D CNN are defined by the state equations

$$\begin{aligned} \dot{x}_{i,j} &= g(x_{i,j}, z_{i,j}, u_{i,j}(t), I^s_{i,j}) \\ y_{i,j} &= f(x_{i,j}) \end{aligned} \tag{2.12}$$

with state vector $x_{i,j} \in \mathbb{R}^{m_x}$, output vector $y_{i,j} \in \mathbb{R}^{m_y}$, threshold $z_{i,j} \in \mathbb{R}^{m_z}$ and input vector $u_{i,j} \in \mathbb{R}^{m_u}$ of the cell $\mathfrak{C}_{i,j}$. $I^s_{i,j} \in \mathbb{R}^{m_I}$ is a synaptic law of the $(i,j)-$th cell $\mathfrak{C}_{i,j}$. $f : \mathbb{R}^{m_x} \to \mathbb{R}^{m_y}$ is nonlinear output function and $g : \mathbb{R}^{m_x} \times \mathbb{R}^{m_z} \times \mathbb{R}^{m_u} \times \mathbb{R}^{m_I} \to \mathbb{R}^{m_x}$.

The synaptic law defines the coupling between the considered cell $\mathfrak{C}_{i,j}$ and all

the cells within the prescribed sphere of influence $\mathcal{S}_{i,j}$

$$\begin{aligned} I^s_{i,j} = \sum_{k,l \in \mathcal{S}_{i,j}} \Big\{ & \hat{A}_{k-i,l-j} x_{k,l} + A_{k-i,l-j} y_{k,l} + B_{k-i,l-j} u_{k,l} \\ & + A^\tau_{k-i,l-j} y_{k,l}(t-\tau) + A^n_{k-i,l-j}(x_{i,j}, x_{k,l}) \\ & + B^\tau_{k-i,l-j} u_{k,l}(t-\tau) + B^n_{k-i,l-j}(u_{i,j}, u_{k,l}) \Big\}. \end{aligned} \tag{2.13}$$

The symbols $k,l \in \mathcal{S}_{i,j}$ denote the indices of all the cells $\mathfrak{C}_{k,l}$ lying inside $\mathcal{S}_{i,j}$. The synaptic law is uniquely specified by synaptic matrices $\hat{A}_{\hat{k},\hat{l}} \in \mathbb{R}^{m_I \times m_x}$, $A_{\hat{k},\hat{l}} \in \mathbb{R}^{m_I \times m_y}$, $B_{\hat{k},\hat{l}} \in \mathbb{R}^{m_I \times m_u}$, $A^\tau_{\hat{k},\hat{l}} \in \mathbb{R}^{m_I \times m_y}$, $B^\tau_{\hat{k},\hat{l}} \in \mathbb{R}^{m_I \times m_u}$ and functions $A^n_{\hat{k},\hat{l}} : \mathbb{R}^{m_x} \times \mathbb{R}^{m_x} \to \mathbb{R}^{m_I}$, $B^n_{\hat{k},\hat{l}} : \mathbb{R}^{m_u} \times \mathbb{R}^{m_u} \to \mathbb{R}^{m_I}$ where $\hat{k},\hat{l} \in \{-r, -r+1, \cdots, r-1, r\} \setminus \{0\}$. Like 1-D CNN, these matrices are listed as entries of state template $\hat{A}$, feedback template A, feedforward template (or *control template*) B, delay-type feedback template A^τ, delay-type feedforward template B^τ, nonlinear feedback template $A^n(x_{i,j}, x_{k,l})$ and nonlinear feedforward template $B^n(u_{i,j}, u_{k,l})$, respectively. For instance the synaptic matrices $A_{\hat{k},\hat{l}}$ are listed in a feedback template

$$A = \begin{bmatrix} A_{-r,-r} & \cdots & A_{-r,r} \\ A_{-r+1,-r} & \cdots & A_{-r+1,r} \\ \cdot & \cdots & \cdot \\ \cdot & \cdots & \cdot \\ \cdot & \cdots & \cdot \\ A_{r,-r} & \cdots & A_{r,r} \end{bmatrix}. \tag{2.14}$$

The most common boundary conditions for 2-D CNNs are:

1. *Fixed (Dirichlet) Boundary Condition:*

$$x_{i,0} = E_1, \quad i \in \{1,2,\cdots,N\} \tag{2.15}$$
$$x_{i,M+1} = E_2, \quad i \in \{1,2,\cdots,N\} \tag{2.16}$$
$$x_{0,j} = E_3, \quad j \in \{1,2,\cdots,M\} \tag{2.17}$$
$$x_{N+1,j} = E_4, \quad j \in \{1,2,\cdots,M\} \tag{2.18}$$

   where $E_1, E_2, E_3, E_4 \in \mathbb{R}$

2. *Zero-Flux (Neumann) Boundary Condition:*

$$x_{i,0} = x_{i,1}, \quad i \in \{1,2,\cdots,N\} \tag{2.19}$$
$$x_{i,M+1} = x_{i,M}, \quad i \in \{1,2,\cdots,N\} \tag{2.20}$$

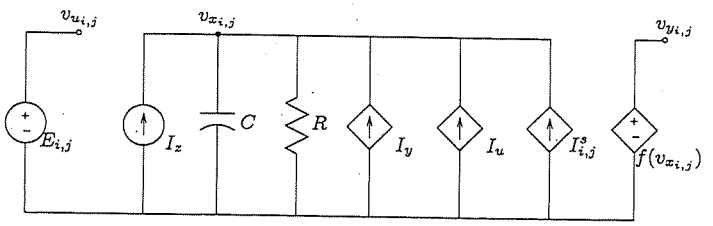


Figure 2.3: Example of a circuit cell for the Chua-Yang model. The node voltages $v_{x_{i,j}}$, $v_{y_{i,j}}$ $v_{u_{i,j}}$ are called the state, output and input of the cell, respectively. C is a linear capacitor; R is a linear resistor; $E_{i,j}$ is an independent voltage source; I_z is an independent current source; $f(v_{x_{i,j}})$ is a nonlinear voltage controlled voltage sources; I_y, I_u and $I^s_{i,j}$ are linear voltage controlled current sources with characteristics $I_y = a_{0,0}v_{y_{i,j}}$, $I_u = b_{0,0}v_{u_{i,j}}$ and the synaptic law of the model (2.28), respectively.

$$x_{0,j} = x_{1,j}, \qquad j \in \{1, 2, \cdots, M\} \tag{2.21}$$
$$x_{N+1,j} = x_{N,j}, \qquad j \in \{1, 2, \cdots, M\} \tag{2.22}$$

3. *Periodic (Toroidal) Boundary Condition:*

$$x_{i,0} = x_{i,M}, \qquad i \in \{1, 2, \cdots, N\} \tag{2.23}$$
$$x_{i,M+1} = x_{i,1}, \qquad i \in \{1, 2, \cdots, N\} \tag{2.24}$$
$$x_{0,j} = x_{N,j}, \qquad j \in \{1, 2, \cdots, M\} \tag{2.25}$$
$$x_{N+1,j} = x_{1,j}, \qquad j \in \{1, 2, \cdots, M\}. \tag{2.26}$$

2.2 CNN models

2.2.1 Chua-Yang CNN model

The original model of CNN was introduced by Chua and Yang [48]. Since then, stability analysis of this model has been extensively studied. The main results can be found in references [225, 10, 49]. Furthermore, bifurcation phenomena and chaotic behavior have been investigated and a chaotic attractor has been observed in its dynamics by Zou and Nossek [326]. The Chua-Yang model is still a widely used model from which a large variety of dynamical behaviors can be observed despite its simple structure.

The cell dynamics of the Chua-Yang CNN model are governed by the following

state equations

$$\begin{aligned} \dot{x}_{i,j} &= -x_{i,j} + a_{0,0}y_{i,j} + b_{0,0}u_{i,j} + z + I^s_{i,j} \\ y_{i,j} &= f(x_{i,j}) = \frac{1}{2}(|x_{i,j}+1| - |x_{i,j}-1|) \end{aligned} \tag{2.27}$$

where $x_{i,j}, y_{i,j}, u_{i,j}, z, I^s_{i,j} \in \mathbb{R}$ (firts order cells).

The synaptic law of the model is given by

$$I^s_{i,j} = \sum_{k,l \in \mathcal{S}_{i,j}} a_{k-i,l-j}y_{k,l} + b_{k-i,l-j}u_{k,l}. \tag{2.28}$$

A fixed boundary condition was proposed by Chua and Yang [48]. The CNN is clamped at its ends to 0. But, the other boundary conditions can be also used. A typical example of a cell for the Chua-Yang CNN model is shown in Figure 2.3.

2.2.2 State controlled CNN (SC-CNN) model

In the Chua-Yang CNN model, the synaptic law is obtained through the inputs and nonlinear outputs only. In [5], Arena *et al.* have removed this restriction by introducing an additional input which directly depend on on the states of the cells in a prescribed sphere of influence. The authors addressed this model for realizing Chua's circuit [5] and also n-double scroll attractors [6]. We will present these results in Section 3.2.1. A cell of a SC-CNN is governed by the following state equations

$$\begin{aligned} \dot{x}_{i,j} &= -x_{i,j} + \hat{a}_{0,0}x_{i,j} + a_{0,0}y_{i,j} + b_{0,0}u_{i,j} + z + I^s_{i,j} \\ y_{i,j} &= f(x_{i,j}) = \frac{1}{2}(|x_{i,j}+1| - |x_{i,j}-1|) \end{aligned} \tag{2.29}$$

where $x_{i,j}, y_{i,j}, u_{i,j}, z, I^s_{i,j} \in \mathbb{R}$.

The synaptic law of the model is given by

$$I^s_{i,j} = \sum_{k,l \in \mathcal{S}_{i,j}} \hat{a}_{k-i,l-j}x_{k,l} + a_{k-i,l-j}y_{k,l} + b_{k-i,l-j}u_{k,l}. \tag{2.30}$$

A specific boundary condition is not described.

2.2.3 Full-range CNN model

CNN VLSI implementation is required to modify the Chua-Yang model into a Full-range CNN model [204]. This new model, allowing all cell variables to have the same variation ranges and, hence, better area and power consumption figures. In [79], it is proved that symmetric Full-range CNNs are stable and that there exists a one-to-one correspondence between the equilibrium points of Chua-Yang CNNs and Full-range CNNs. Most recently, Corito and Gilli [51] have shown that the sufficient conditions under which CNN-convergence can be proved are identical for both Chua-Yang CNNs and Full-range CNNs. However, the analysis of global dynamic behaviors shows that there exist sets of identical parameters for which they present a qualitatively different dynamic behavior [51].

The cell dynamics of Full-range CNN are described by the following mathematical model

$$\begin{aligned} \dot{x}_{i,j} &= -g(x_{i,j}) + a_{0,0}y_{i,j} + b_{0,0}u_{i,j} + z + I^s_{i,j} \\ y_{i,j} &= f(x_{i,j}) = \frac{1}{2}(|x_{i,j}+1| - |x_{i,j}-1|) \end{aligned} \tag{2.31}$$

with $x_{i,j}, y_{i,j}, z, I^s_{i,j} \in \mathbb{R}$ and

$$g(x_{i,j}) = \begin{cases} m(x_{i,j}-1)+1 & x_{i,j} < -1 \\ x_{i,j} & |x_{i,j}| \leq 1 \\ m(x_{i,j}+1)-1 & x_{i,j} > 1 \end{cases} \tag{2.32}$$

where m is supposed to be large enough for approximating the non-linear characteristic shown in Figure 2.4. Taking $m = 1$, Eq. (2.31) reduce to cell dynamics of the Chua-Yang CNN model. The synaptic law of the model is given by

$$I^s_{i,j} = \sum_{k,l \in \mathcal{S}_{i,j}} a_{k-i,l-j}y_{k,l} + b_{k-i,l-j}u_{k,l}. \tag{2.33}$$

CNNs of size 20×22, 64×64 and 128×128 have been implemented using this model in the ACE400 [70], ACE4k [161] and ACE16k [162] chips, respectively. While ACE4k allows fixed, zero-flux and periodic boundary conditions, the others only allow fixed and zero-flux boundary conditions.

2.2.4 Reaction-diffusion CNN model

An important class of partial differential equations (PDEs) are reaction-diffusion type PDEs

$$\frac{\partial U(\bar{x},\bar{y},t)}{\partial t} = h(U(\bar{x},\bar{y},t)) + \nabla^2 U(\bar{x},\bar{y},t) \tag{2.34}$$

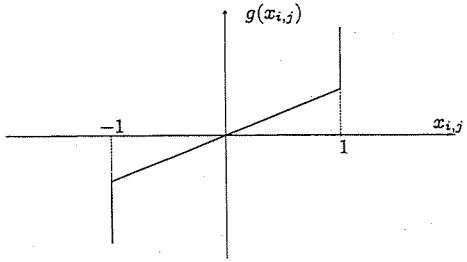


Figure 2.4: The proposed nonlinear function for $g(x_{i,j})$ in the Full-range CNN model [204, 161].

where $U = [U_1, U_2, \cdots U_m]^T \in \mathbb{R}^m$, $h : \mathbb{R}^m \to \mathbb{R}^m$ and

$$\nabla^2 U_i = \frac{\partial^2 U_i}{\partial \bar{x}^2} + \frac{\partial^2 U_i}{\partial \bar{y}^2} \quad i = 1, 2, \cdots, m \tag{2.35}$$

is the Laplacian operator in $\mathbb{R}^2$ ($\bar{x}$ and $\bar{y}$ are designated as spatial axes). For solving such an equation, spatial discretization has to be applied. The PDE is transformed into a system of ODEs which is identified as the state equations of a CNN with an appropriate synaptic law. The discretization in space is made in equidistant steps h in both direction ($\Delta\bar{x} = \Delta\bar{y} = h$) on the $N \times M$ grid. So $U(\bar{x},\bar{y},t)$ is mapped into a CNN layer such that state variable $x_{i,j}(t)$ of a CNN cell $\mathfrak{C}_{i,j}$ is associated with $U(ih, jh, t)$ where $i \in \{1, 2, \ldots, N\}$ and $j \in \{1, 2, \ldots, M\}$ [41, 213].

Based on the Taylor-series expansion of $U(\bar{x},\bar{y},t)$ the Laplacian operator is approximated by

$$\nabla^2 U \sim \frac{1}{h^2}[U(\bar{x}+h,\bar{y}) + U(\bar{x}-h,\bar{y}) + U(\bar{x},\bar{y}+h) + U(\bar{x},\bar{y}-h) - 4U(\bar{x},\bar{y})]. \tag{2.36}$$

Hence a cell of reaction-diffusion CNN is governed by the following state equation

$$\dot{x}_{i,j} = h(x_{i,j}) + A_{0,0}x_{i,j} + I^s_{i,j} \tag{2.37}$$

where $x_{i,j} \in \mathbb{R}^m$, $I^s_{i,j} \in \mathbb{R}^m$ and $A_{00} = -\frac{4}{h^2}I_{m\times m}$. The synaptic law of the model is given by

$$I^s_{i,j} = \sum_{k,l \in \mathcal{S}_{i,j}} A_{k-i,l-j}x_{k,l} \tag{2.38}$$

with a 3×3 sphere of influence $\mathcal{S}_{i,j}(1)$. The synaptic matrices are $A_{-1,0} = a_{0,-1} = A_{0,1} = A_{1,0} = \frac{1}{h^2} I_{m \times m}$, $A_{-1,-1} = A_{-1,1} = A_{1,-1} = A_{1,1} = 0_{m \times m}$.

The model has been used as a paradigm for generating autowaves, spiral waves, scroll waves and spatial-temporal chaos [40]. For instance in [185], Chua's circuit has been used as a CNN cell for generating travelling autowaves, spiral waves and Turing patterns.

2.2.5 Generalized CNN models

In [100], a generalized CNN model has been obtained by modifying the connection topology and cell dynamics. This generalized model is a multi-dimensional, multilayer network of arbitrary order dynamical cells with linear templates. The other generalized CNN model has been developed in [211] by choosing nonlinear input and feedback functions which are defined by nonlinear templates (A^n and B^n) in the Chua-Yang model.

2.2.5.1 A generalized CNN model: nonlinear and delay CNNs

Although the CNN, which is based on the Chua-Yang model, is a nonlinear system it has been referred to as a linear CNN (and also its templates A and B have been referred to as *linear feedback* and *feedforward* templates) since a generalized model allows nonlinear templates as introduced by Roska and Chua [211].

The cell dynamics of the generalized CNN model in [211] are described by the following state equations

$$\begin{aligned} \dot{x}_{i,j} &= -x_{i,j} + a^n_{0,0}(x_{i,j}, x_{k,l}) + b^n_{0,0}(u_{i,j}, u_{k,l}) \\ &\quad + a^\tau_{0,0} y_{i,j}(t-\tau) + b^\tau_{0,0} u(t-\tau) + z + I^s_{i,j} \\ y_{i,j} &= f(x_{i,j}) = \frac{1}{2}(|x_{i,j}+1| - |x_{i,j}-1|) \end{aligned} \tag{2.39}$$

with $x_{i,j}, y_{i,j}, z, I^s_{i,j}, \tau \in \mathbb{R}$. Possibly $\tau = \tau_{i,j}$. The synaptic law of this generalized model is given by

$$\begin{aligned} I^s_{i,j} = \sum_{k,l \in \mathcal{S}_{i,j}} &\{a^n_{k-i,l-j}(x_{i,j}, x_{k,l}) + b^n_{k-i,l-j}(u_{i,j}, u_{k,l}) \\ &+ a^\tau_{k-i,l-j} y_{i,j}(t-\tau) + b^\tau_{k-i,l-j} u(t-\tau)\}. \end{aligned} \tag{2.40}$$

In the case when $a^n_{k-i,l-j}$ and $b^n_{k-i,l-j}$ are linear functions and $a^\tau_{k-i,l-j}$ and $b^\tau_{k-i,l-j}$ are equal to zero, we have the Chua-Yang model with linear templates.

As a results of this generalization, *delay type CNN* and *nonlinear CNN* have been introduced by Roska and Chua [211] for $a^n_{k-i,l-j} = b^n_{k-i,l-j} = 0$ and $a^\tau_{k-i,l-j} = b^\tau_{k-i,l-j} = 0$, respectively. In [211] the authors also discussed non-uniform grid CNN architecture having more than one type of cell dynamics and named as a *non-uniformed processor CNN* (NUP-CNN). Furthermore, a larger than 3×3 sphere of influence ($\mathcal{S}_{i,j}(1)$) has been referred to as a *multiple neighborhood size CNN* (MNS-CNN) [211].

From an application point of view, while delay-type CNNs find applications in detection of moving objects [210, 216], nonlinear CNNs are useful for image processing tasks [128].

2.2.5.2 Another generalized CNN model

As we have mentioned, Güzeliş and Chua [100] developed a generalized model for CNN which is a multi-dimensional multilayer network. This generalized CNN is a very general network model in view of to the connection topology and the dimension.

Each cell $\mathfrak{C}_{i,j}$ in the model is a $m_{i,j}$th order nonlinear dynamical system. The cell decomposes into three subsystems. The first subsystem in (2.43) which is a multi-input single output system, contains the synaptic input to the cell. The second subsystem in (2.41) is a single-input ($I^s_{i,j}$) single-output ($\xi_{i,j}$) $m_{i,j}$th order linear dynamical system and it is fed by the first subsystem. The third subsystem (2.42) contains the nonlinearity $f(\cdot)$ of the cell.

The cell dynamics of the generalized CNN model in [100] are described by the following state equations

$$\begin{aligned} \dot{x}_{i,j} &= A_{i,j} x_{i,j} + B_{i,j}(a^\tau_{0,0} y_{i,j}(t-\tau_1) + b^\tau_{0,0} u_{i,j}(t-\tau_2) + I^s_{i,j}) + z_{i,j} \\ \xi_{i,j} &= C^T_{i,j} x_{i,j} + h_{i,j} I^s_{i,j} \end{aligned} \tag{2.41}$$

and

$$y_{i,j} = f_{i,j}(\xi_{i,j}) \tag{2.42}$$

where $A \in \mathbb{R}^{m_{i,j} \times m_{i,j}}$, $B_{i,j}, C_{i,j} \in \mathbb{R}^{m_{i,j}}$, $h_{i,j}, z_{i,j}, y_{i,j}, u_{i,j}, \xi_{i,j}, a^\tau_{0,0}, b^\tau_{0,0} \in \mathbb{R}$, $x_{i,j} \in \mathbb{R}^{m_{i,j}}$. Possibly $\tau_1 = \tau_{1_{i,j}} \in \mathbb{R}$ and $\tau_2 = \tau_{2_{i,j}} \in \mathbb{R}$. The synaptic law of this generalized model is given by

$$I^s_{i,j} = \sum_{k,l \in \mathcal{S}_{i,j}} a^\tau_{k-i,l-j} y_{i,j}(t-\tau_1) + b^\tau_{k-i,l-j} u_{i,j}(t-\tau_2) \tag{2.43}$$

where $a^\tau_{k-i,l-j}$ and $b^\tau_{k-i,l-j} \in \mathbb{R}$.

In order to find the conditions ensuring the global stability of an isolated cell of this generalized model with zero input, Güzeliş and Chua [100] linked the problem with Lur'e system. The isolated cell is in fact a Lur'e system and it belongs to a sector condition for many nonlinear output functions ($f(\cdot)$). Therefore, we can consider this type of CNN a network of locally coupled identical Lur'e systems placed on a 2-D grid. This property has been exploited by NL_q theory [248] as a general framework for the analysis and design of such network.

2.2.5.3 A generalized CNN based on Chua's circuit

A special type of a generalized CNN is the chaotic CNN [99] which is later called a $(f(x), \dot{x})$ coupled chaotic CNN in order to distinguish it from other recently proposed chaotic CNN architectures [126]. Each cell in the chaotic CNN is a Chua's circuit which exhibits a variety of nonlinear dynamic behaviors, when it is isolated from the synaptic input and the external input. Chua's circuit (Section 3.1) and isolated chaotic CNNs (Section 3.1.2) are studied in the next Chapter.

The cell dynamics of the chaotic CNN are described by (2.41) with $m_{i,j} = 3$,

$$A_{i,j} = \begin{bmatrix} -\alpha(1+\delta) & \alpha & 0 \\ 1 & -1 & 1 \\ 0 & -\beta & 0 \end{bmatrix}, B_{i,j} = \begin{bmatrix} \alpha \\ 0 \\ 0 \end{bmatrix}, C_{i,j} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, h_{i,j} = 0$$

where $\alpha, \delta, \beta \in \mathbb{R}$. $f_{i,j}(\xi) = g(\xi)$ and the synaptic law are given in (3.30) and (2.43), respectively.

2.3 CNN universal machine: a visual microprocessor

Since its invention in 1988, the CNNs subject has received considerable attention from mathematicians, physicists, neurobiologists and also engineers. In this way CNNs have become a new paradigm [45]. The CNN paradigm has provided an ideal framework for programmable analog array computing. It means that the CNN can be used as a programmable device where the instructions are represented by the templates. This was the key idea in proposing the CNN universal machine (CNN-UM) which is the first algorithmically programmable analog array computer [212]. Another aspect which distinguishes the CNN-UM from a conventional digital computer is the *dual computing* capability. Dual computing combines analog array processing with logic operations, therefore

named as *analogic* (which is a contraction of analog and logic computation) computing by Roska and Chua [212]. Universality of the CNN Universal Machine has been proved by Roska & Chua [212] and Crounse & Chua [54] in the sense of Turing which meaning that a (Turing) universal machine is able to realize any recursive function. In order to avoid Input/Output connections with the external world on the CNN-UM, Roska and Chua [212] proposed the integration of an array of sensors on the same chip.

The first physical implementation of the CNN-UM proved the physical realizability of the architecture [58, 71]. The availability of good CMOS phototransduction devices and the possibility to realize linear and nonlinear processing functions with simple CMOS circuitry recently made it possible to integrate the sensory and the processing circuitry on the same semiconductor substrate. As a result of this technological development, Analogic Cellular Engines (ACE4k [163] and ACE16K [162]) which are designed based on the CNN-UM architecture, have been integrated with optical sensors. Therefore, ACE4k and ACE16k are actually a focal plane *visual microprocessor* because the picture is projected and sensed by photo sensors fabricated on the surface of the chip. Furthermore they are capable of realizing a large variety of image related spatial-temporal operations and algorithms through the execution of a suitable sequence of instructions (or templates) [214]. Table 2.1 compares some of the recently developed CNN-UM chips. Furthermore, a mammalian retina model based on [205] can be characterized by two coupled sets of 2-D nonlinear differential equations shown in [16]. This model has been implemented on a silicon chip (CACE1k) [23] based on a CNN-UM architecture with some extensions of core processing unit (a CNN) from a single layer analog programmable array processor to a two layers analog programmable arrays processor (a multilayer CNN).

Besides the development of the dedicated VLSI hardware realizations of CNN-UMs, a computational infrastructure which is the CNN chip prototyping system (CCPS) has also emerged, interfacing this analog array computer to digital systems [217]. The main goal was to provide user friendly tools and hardware interfacing elements and putting most of the sophistication on the chips. The CCPS was designed and implemented to measure and program a CNN-UM chip in a standard PC environment by using a high-level language (Alpha) and as well as a low-level language (Analogic Machine Code (AMC)). Recently a high performance CCPS has been introduced for the professional computational environment of ACE4k and ACE16k [320].

From an application point of view, the CNN universal chip allows to use the algorithm introduced as an efficient pre-processing procedure for many interesting image/video applications such as skeletonizing [274], segmentation [8],

Reference	[70]	[194]	[163]	[162]
Array Size	20×22	48×48	64×64	128×128
Pixel Format	B/W	B/W	Analog	Analog
Photo Sensor	Yes	No	Yes	Yes
Res. for Weights	8	6	8	8
Program Memory	8 Templ.	1 Templ.	32 Templ.& 64 Dig. Inst.	32 Templ.& 4096 Dig. Inst.
Memory per Cell	4 B/W	2 In/ 2 Out Registers	4 B/W & 4 Analog	8 Analog
Speed (XPS)	0.13T	7.65T	0.04T	0.33T
Technolog CMOS	$0.8\mu m$	$0.5\mu m$	$0.5\mu m$	$0.35\mu m$
# Transistors			$\sim 1,000,000$	3,748,170
Cells/mm^2	27.5	295	82	180

Table 2.1: Summary and comparison of the recent CNN-UM chip implementations. (B/W : Black/White, Res. for Weight: It refers to the number of bits used to defined a synaptic coefficient, XPS: Analog operations per second is an equivalent measurement indicating the number of analog arithmetic operations like addition, subtraction, multiplication and division, Temp: Template).

watermarking [309], the binary and gray-scale morphological erosion, dilation, and reconstruction operators [321] and many others [46]. For these operations, there is already an image processing library which contains a collection of designed templates [260]. The applications of CNNs and CNN-UMs, which are developed in this thesis, will be discussed in Chapter 5.

2.4 New research aspects for CNNs

2.4.1 Wave computing algorithm

Recently, Roska [206] has introduced Spatial-temporal Instruction Set Computer (StISC) whose elementary instructions are continuous nonlinear spatial-temporal waves defined on a cellular nonlinear network via templates. The CNN-UM architecture is a universal model of StISCs [206, 207]. Basically StISCs perform signal processing operations via waves.

Contrary to conventional digital computer instructions, spatial-temporal instructions for image processing are continuous in time and defined on *image flows*

$$\Phi(t) : \{\varphi_{i,j}(t), t \in T = [0, t_1], 1 \leq i \leq N, 1 \leq j \leq M\} \tag{2.44}$$

where t_1 is the time duration, $\varphi_{i,j}(t)$ may represent an input, state or output of the cell in the array and each cell has a pixel intensity $p_{i,j}$ with $|p_{i,j}| \leq p_{\max}$ where $p_{i,j} in \mathbb{R}$ and $p_{\max} \in \mathbb{R}$ (see Figure 2.5). The design of algorithms on StISCs requires a different way of thinking in order to take advantage of the computational power of StISCs. Today, this computing approach is called *wave computing* and StISCs are also named *wave computers*.

The elementary instruction is defined as

$$\Phi_{\text{output}}(t) = \Psi(\Phi_{\text{input}}(t), P, \delta, I^s_{i,j}) \quad t \in T = [0, t_1] \tag{2.45}$$

where Ψ is a function on image flow, P is a picture defining the initial state, δ represents the boundary conditions and $I^s_{i,j}$ (2.13) is a synaptic law [206]. Figure 2.5 shows the outputs $\Phi_1(t_1), \Phi_2(t_2), \Phi_3(t_3)$ ($\Phi(t_n)$ are pictures (or snapshots) at time t_n where $n = 1, 2, 3$ of three consecutive runs of an instructions on the array. The outputs can be a mask M (a matrix) where $M = [m_{i,j}]$ where $m_{i,j} \in \{-p_{\max}, p_{\max}\}$.

Spatial-temporal instructions are grouped into two categories: equilibrium and non-equilibrium spatial-temporal type instructions. If after a time duration t_n the output is not settled, i.e. if there exists at least one $\dot{\varphi}_{i,j} \neq 0$ then one has

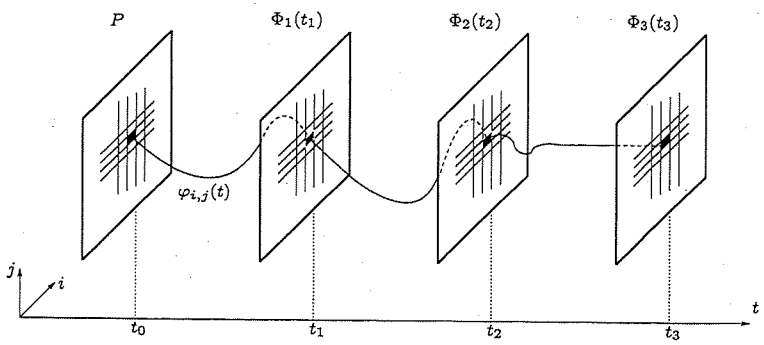


Figure 2.5: An illustration of a wave computing algorithm whose input P and output $\Phi_3(t_3)$ are an image at time $t = t_0$ and t_3 respectively. The algorithm has three instructions which run consecutively.

non-equilibrium spatial-temporal type instructions. For instance, in Figure 2.5, the output of the instruction Φ_3 for the cell $\mathbf{C}_{i,j}$ is settled.

Essential features of wave computing on a StISC are [206]:

- non-equilibrium type spatial-temporal elementary instructions,
- spatial logic instruction on masks,
- spatial-temporal combination of image flows,
- algorithms using the above three types of instructions.

In Chapter 4 of [203], it has been shown that a properly controlled wave computing technique can be efficiently used in segmentation, shape and structure detection/recovery tasks. Furthermore, for coupled local minimizers discussed in the next Section, state synchronization is used as an appropriate technique to control wave computing in order to obtain better local minima for a given optimization task.

In fact, many types of naturally occurring spatial-temporal waves are described by nonlinear PDEs. Interestingly, the computer vision community recently started to be interested in PDEs and found interesting properties well adapted to handle classical image processing problems [3, 269, 61]. The relationship between PDEs and CNNs have been already presented by Chua and Yang [48], Roska *et al.* [213] and more recently by Gilli *et al.* in [93].

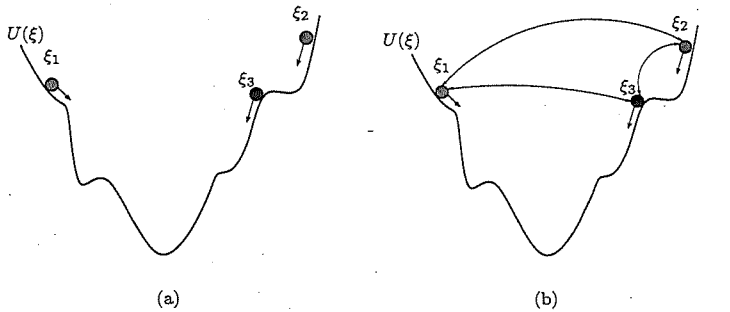


Figure 2.6: (a) Multi-start local optimization; (b) In CLM local minimizers works cooperatively toward finding improved local minima using intraction and information exchange between the local search processes.

2.4.2 Coupled local minimizers

In [258] a method of coupled local minimizers has been recently proposed where local coupling between continuous time local optimization networks has been defined. Interpreted within the CNN framework, cells correspond then to local optimization algorithms and information exchange between the local cells takes place through state synchronization constraints that express that the minimizers should reach the same state asymptotically. It has been shown that the achieved local minima of CLMs can be better than the best results obtained from individual local minimizers [258].

Consider the unconstrained problem of minimization of a twice continuously differentiable cost function

$$\min_{\xi \in \mathbb{R}^n} U(\xi). \tag{2.46}$$

A simple continuous time local optimization algorithm is the gradient system (steepest descent method)

$$\dot{\xi} = -\mu \nabla_\xi U(\xi) \tag{2.47}$$

with $\mu \in \mathbb{R}^+$ the step size. Since $\nabla_\xi U(\xi)$ is minimized, the time evolution of ξ will result in $U(\xi)$ becoming smaller and smaller as time goes on. Therefore, the process of seeking for an equilibrium of the gradient system of a cost function $U(\xi)$ actually yields local minimization of this cost function.

Instead of running such local optimization methods in a multi-start fashion (i.e. independently from each other for different initial conditions (See Figure

2.6(a))), in coupled local minimizers an ensemble of local minimizers has been proposed (see Figure 2.6(b)) that interacts through state synchronization constraints. This implicitly incorporates a mechanism of master-slave dynamics as is known in chaos synchronization methods which will be discussed in Chapter 4.

A group consisting of a number of q minimizers is considered for which one aims at minimizing the average energy cost

$$\langle U \rangle = \frac{1}{q} \sum_{i=1}^{q} U[\xi_i] \tag{2.48}$$

subject to pairwise state synchronization [258]

$$\min_{\xi_i \in \mathbb{R}^n} \langle U \rangle \text{ such that } \quad \xi_i - \xi_{i+1} = 0 \tag{2.49}$$

with individual states $\xi_i \in \mathbb{R}^n$ for $i = 1, 2, ..., q$ and boundary conditions $\xi_0 = \xi_q$, $\xi_{q+1} = \xi_1$ (in the case of a 1-D CNN configuration). The synchronization constraints have to be achieved in an asymptotic sense, i.e. the individuals have to reach the same final state.

One defines the augmented Lagrangian [82]

$$\mathcal{L}(\xi_i; \lambda_i) = \frac{\eta}{q} \sum_{i=1}^{q} U[\xi_i] + \frac{1}{2} \sum_{i=1}^{q} \gamma_i \left\| \xi_i - \xi_{i+1} \right\|_2^2 + \sum_{i=1}^{q} \langle \lambda_i, [\xi_i - \xi_{i+1}] \rangle \tag{2.50}$$

with Lagrange multipliers $\lambda_i \in \mathbb{R}^n$ $(i = 1, 2, ..., q)$ where $\langle \cdot, \cdot \rangle$ denotes the inner product between two vector. The penalty factors γ_i emphasize the importance of each of the soft synchronization constraints.

From (2.50), one obtains the Lagrange programming network [322]

$$\left\{ \begin{array}{rcl} \dot{\xi}_i & = & -\nabla_{\xi_i} \mathcal{L}(\xi_i; \lambda_i) \\ \dot{\lambda}_i & = & \nabla_{\lambda_i} \mathcal{L}(\xi_i; \lambda_i) \, , \, i = 1, 2, ..., q. \end{array} \right. \tag{2.51}$$

This gives the following array of coupled local minimizers

$$\left\{ \begin{array}{rcl} \dot{\xi}_i & = & -\frac{\eta}{q} \nabla_{\xi_i} U[\xi_i] + \gamma_{i-1}[\xi_{i-1} - \xi_i] - \gamma_i [\xi_i - \xi_{i+1}] \\ & & +\lambda_{i-1} - \lambda_i \\ \dot{\lambda}_i & = & \xi_i - \xi_{i+1} \, , \, i = 1, 2, ..., q \end{array} \right. \tag{2.52}$$

with learning rate η. This array consists of a number of q coupled nonlinear cells and can be considered as a CNN with following cell dynamic

$$\dot{x}_i = \begin{bmatrix} -\frac{\eta}{q} \nabla_{\xi_i} U[\xi_i] \\ 0 \end{bmatrix} + \begin{bmatrix} -(\gamma_{i-1} + \gamma_i) I_n & -I_n \\ I_{n \times n} & 0_n \end{bmatrix} x_i + I_i^s \tag{2.53}$$

where $x_i = [\xi_i \ \lambda_i]^T, I_i^s \in \mathbb{R}^{2n}$. The synaptic law of this CLM is given by

$$I_i^s = \sum_{k \in \mathcal{S}_i(1)} \hat{A}_{k-i} x_k \tag{2.54}$$

with a sphere of influence $\mathcal{S}_i(1)$. The synaptic coefficients are

$$\hat{A}_{-1} = \begin{bmatrix} \gamma_{i-1} I_n & I_n \\ 0_{n \times n} & 0_{n \times n} \end{bmatrix} \text{ and } \hat{A}_1 = \begin{bmatrix} \gamma_i I_n & 0_{n \times n} \\ -I_n & 0_{n \times n} \end{bmatrix}.$$

A periodic boundary condition was imposed for this CLM-CNN model [258]. This method exploits the spatial-temporal behavior of an 1-D CNN to obtain a better local minima which appears when the state synchronization occurs. Therefore, CLMs perform a form of wave computing algorithm which is controlled by the state synchronization.

This method has been successfully applied so far to e.g. the training of multilayer perceptron neural networks, global optimization problems in finite element modeling for civil engineering applications and optimization of Lennard-Jones clusters [258]. Optimal cooperative search is possible where the coupling constants are re-scheduled after solving linear programs in the coupling constants [258]. The performance can be further enhanced also by re-numbering the cells, motivated by principles of small-world networks [280]. In this thesis we introduce a new method which is *coupled chaotic annealing* in order to improve the performance of CLM, discussed in Section 5.2.2.

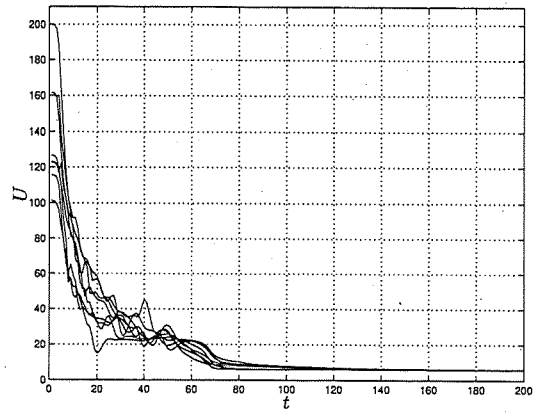
Example 2.1 We illustrate the principle of CLMs here on the Rastrigin function which is a known test function [239]

$$U(\xi) = \frac{a}{2n} \xi^T \xi + 4n - 4n \prod_{i=1}^{n} \cos(\omega e_i \xi) \tag{2.55}$$

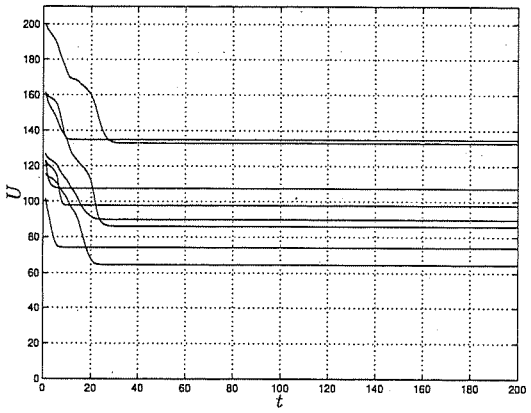
where $\xi \in \mathbb{R}^n$ and

$$e_i = \underbrace{[0 \, 0 \ldots 0}_{i-1} \ 1 \ \underbrace{0 \ldots 0]}_{n-i}$$

with $a = 1$ and $\omega = 1$ in $n = 5$ dimensional search spaces. The global minimum for this problem is known and located at the origin with $U(0) = 0$. Figure 2.7 shows simulation examples on a comparision between multi-start local optimization (i.e. steepest descent method) versus the CLM approach. In these comparions the initial states have been randomly generated in $[-25 \ 25]^n$ where the same initial state have been taken for the uncoupled gradient systems and several runs of the networks have been done (see Figure 2.7(b)). Figure 2.7(a) illustrates when these GNs are coupled with CLMs technique. A number of $q = 8$ minimizers has been taken and $\gamma_i = 5 \ \forall i$. The solution from CLMs improves with respect to multi-start local optimization. △



(a)



(b)

Figure 2.7: (a) Illustration of CLM convergence with $q = 8$ minimizers. (b) Multi-start local optimization for the same problem as in Figure 2.7(a) with the same initial conditions but without coupling.

2.5 Conclusion

While one of the purposes is to explore the computational capability of CNNs for information processing tasks, another is to simulate many phenomena that have been described in the vast literature on complexity. However, today we see that these two purposes have been incorporated under the umbrella of wave computing. Wave computing basically exploits spatial-temporal dynamic behaviors on the CNN array to process information. This new computing technique has already opened new research directions for CNNs such as computing and computer complexity on flows, qualitative theory of spatial-temporal dynamics and spatial-temporal algorithmic techniques [209]. Although this way of computing is one of the most complex task for a digital computer, it is an elementary operation for a CNN based wave computer. Therefore, such wave computers are going to change the complexity scenario [208]. Physical implementations of the CNN based analogic array computer with integrated sensors are a main engine for new research aspects.

Chapter 3

Multi-scroll Chaotic and Hyperchaotic Attractors

In this Chapter we discuss multi-scroll chaotic and hyperchaotic attractors. Multi-scroll chaotic attractors find their origins in the double scroll attractor which is a strange attractor of the well-known Chua's circuit. Since then the observation of n-double scroll attractors on the generalized Chua's circuit, multi-scroll attractors have received considerable attention in the nonlinear circuit and system society. We present an overview of known multi-scroll chaotic and hyperchaotic attractors with their circuit implementations.

The Chapter is organized as follows. In Section 3.1 we review Chua's circuit, including determination of the existence and location of chaotic behavior and its circuit realizations. In Section 3.2 we introduce a generalized Chua's circuit. Furthermore, we review the circuit realizations of the generalized Chua's circuit in order to obtain n-scroll attractors. In Section 3.3 families of scroll grid attractors are presented with their simple and systematic circuit realizations. In Section 3.4 we introduce multi-scroll hyperchaotic attractors which include hyperchaotic n-scroll attractors from generalization of a four dimensional circuit and n-scroll hypercubes attractors form unidirectional or diffusive coupled n-scroll attractors. In Section 3.5, we develop scroll maps from $1D$-scroll grid attractors. Finally, in Section 3.6 Lur'e representations of all chaotic and hyperchaotic circuits are given. New contributions are made in Sections 3.2.1.1, 3.2.1.2, 3.3, 3.4.1, and 3.5.

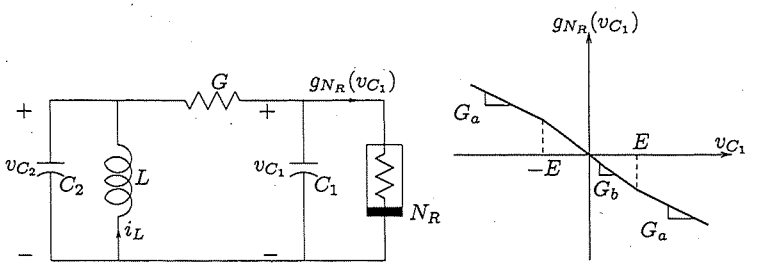


Figure 3.1: Chua's circuit with three segment piecewise-linear function.

3.1 Chua's circuit

Chua's circuit (Figure 3.1) is described by a set of three ordinary differential equations

$$\begin{cases} C_1 \frac{dv_{C_1}}{dt} &= G(v_{C_2} - v_{C_1}) - g_{N_R}(v_{C_1}) \\ C_2 \frac{dv_{C_2}}{dt} &= G(v_{C_1} - v_{C_2}) + i_L \\ L \frac{di_L}{dt} &= -v_{C_2} \end{cases} \tag{3.1}$$

where v_{C_1} and v_{C_2} are the voltages across capacitors C_1 and C_2 respectively, and i_L is the current flowing upward through the inductor [173]. The nonlinear function $g_{N_R}(v_{C_1})$ which defines the $v - i$ characteristic of nonlinear resistor N_R (called *Chua's diode* [129]), is described by the piecewise-linear function (see Figure 3.1):

$$g_{N_R}(v_{C_1}) = G_b v_{C_1} + \frac{1}{2}(G_a - G_b)\left(|v_{C_1} + E| - |v_{C_1} - E|\right). \tag{3.2}$$

One can reduce the number of parameters by normalizing the nonlinear resistor such that its breakpoints are at ∓ 1V instead of $\mp E$ V. Furthermore, Chua's circuit equations (3.1) in normalized dimensionless form are obtained by making the following change of variables : $x = \frac{v_{C_1}}{E}$, $y = \frac{v_{C_2}}{E}$, $z = \frac{i_L}{EG}$, $\alpha = \frac{C_2}{C_1}$, $\beta = \frac{C_2}{LG^2}$, $m_0 = \frac{G_a}{G}$ $m_1 = \frac{G_b}{G}$ and $\tau = \frac{tG}{C_2}$ leading to

$$\begin{cases} \dot{x} &= \alpha(y - x - f(x)) \\ \dot{y} &= x - y + z \\ \dot{z} &= -\beta y \end{cases} \tag{3.3}$$

where

$$f(x) = m_1 x + \frac{1}{2}(m_0 - m_1)\left(|x + 1| - |x - 1|\right). \tag{3.4}$$

Equation (3.3) is dynamically equivalent with (3.1) and called as *Chua's equation* which is described by the following set of four parameters $\{\alpha, \beta, m_0, m_1\}$ [174]. Furthermore, qualitatively similar dynamical behavior has been also observed with the smooth cubic function

$$f(x) = k_0 x + k_1 x^3 \tag{3.5}$$

where $k_0 = -\frac{8}{7}$ and $k_1 = \frac{4}{63}$ in [174]. The rich dynamical behavior of Chua's circuit was first demonstrated by computer simulation in [173]. Then, the first experimental Chua's circuit that confirms the presence of chaos and the bifurcation phenomena was presented by Zhong and Ayron [324, 323]. The second experimental Chua's circuit was reported in [174] and its chaotic attractor is called as the *double scroll attractor* (see Figure 3.2(a)) which occurs in Chua's equation with the neighborhood of the parameters $\{\alpha, \beta, m_0, m_1\}$= $\{9, 100/7, -8/7, -5/7\}$. The *spiral* [69] and the *double-hook* [17] attractors are also strange attractors that occur in Chua's equation for the parameters $\{8.5, 100/7, -8/7, -5/7\}$ and $\{-4.9167, -3.642, -2.5, -0.93\}$, respectively. Figures 3.2(b) and 3.2(c) show phase portraits of these strange attractors, respectively. After all these experimental and computer simulation based confirmations of chaos in Chua's circuit, Chua *et al.* [43] gave the first rigorous proof of the existence of chaotic behavior in Chua's circuit.

In 1990, Chua and Lin [44] introduced a canonical circuit (by adding a linear resistor in series with an inductor) which is canonical in the sense that every continuous three-dimensional odd-symmetric piecewise-linear vector field may be mapped onto the circuit. The circuit is named *Chua's oscillator* in Madan [167]. Chua's oscillator can thus exhibit every dynamical behavior known to be possible in a system described by a continuous odd-symmetric three-region piecewise-linear vector field [47]. For a comprehensive bibliography of papers on Chua's circuit, we refer to Chua's historical review paper [37].

3.1.1 Chaos in Chua's circuit

3.1.1.1 The Shil'nikov method

The chaotic nature of the double scroll attractor was proved by Chua *et al.* [43] by establishing the existence of a homoclinic loop of the saddle focus at the origin and applying the Shil'nikov theorem [234]. Silva [235] also gave another proof using the Shil'nikov theorem. The theory of Shil'nikov provides an analytical tool available for the study of chaotic dynamics. It addresses the complicated dynamics that can be found in nonlinear autonomous systems of the third degree and higher that possess special types of orbits.

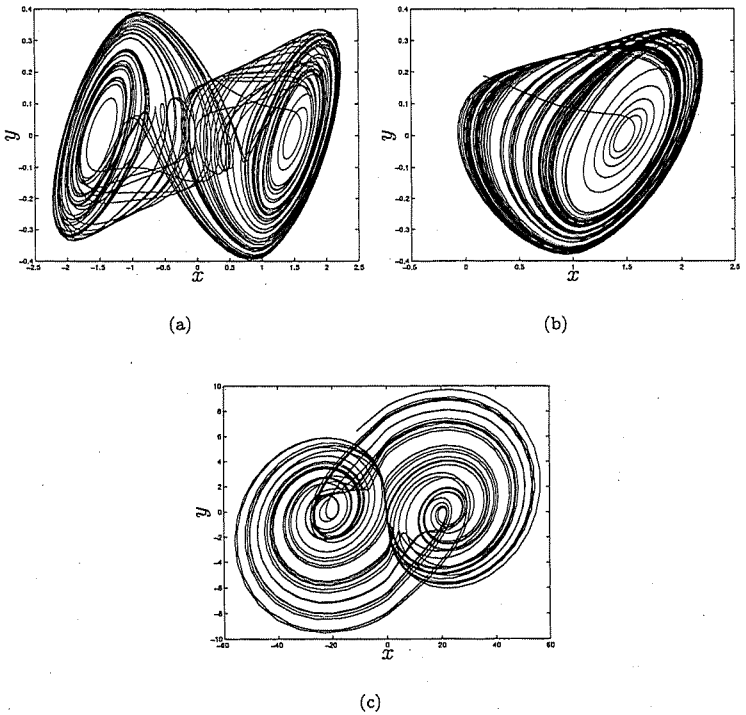


Figure 3.2: Phase portrait of the (a) double scroll, (b) spiral, (c) double hook attractors obtained form Chua's equation.

The system studied by Shil'nikov consisted of a third-order dynamical system

$$\dot{x} = \mathcal{Q}(x), \quad x \in \mathbb{R}^3 \tag{3.6}$$

where the vector field $\mathcal{Q} : \mathbb{R}^3 \to \mathbb{R}^3$ is C^p. The dynamic system (3.6) is said to contain a saddle-focus-type equilibrium point x_e if the eigenvalues of the 3×3 real Jacobian matrix of $\mathcal{Q}(x_e)$ are of the form

$$\gamma, \sigma \pm jw, \quad \sigma\gamma < 0 \text{ and } w \neq 0$$

where σ, γ and w are real.

Figure 3.3 presents the two special orbits that are a necessary condition for the Shil'nikov approach. A *homoclinic orbit* of an equilibrium point is a trajectory $\mathcal{H}$ which tends to the equilibrium point in both forward and reverse time. A *heteroclinic orbit* between the two saddle-focus-type equilibrium points x_{e1} and x_{e2} is a trajectory which tends toward x_{e1} in reverse time and toward x_{e2} in forward time. The formal establishment of the existence of these special orbits is the most difficult part of the method [236]. The system (3.6) is then called a Shil'nikov system if these special orbits based at saddle-focus-type equilibrium points are present in the dynamics. There are two important concepts in the Shil'nikov method which are the Poincaré map and the Smale horseshoe [238]. The method generates discrete dynamics from the Shil'nikov system by using an appropriate Poincaré map (called Shil'nikov map) which is chosen locally with respect to these special orbits. One can show that the Smale horseshoe is embedded in the Shil'nikov map when the Shil'nikov inequality is satisfied. Then there is an uncountable set of saddle type periodic orbits of arbitrarily long period and an uncountable set of bounded nonperiodic orbits in a neighborhood of these special orbits. This behavior is called *horseshoe chaos.*

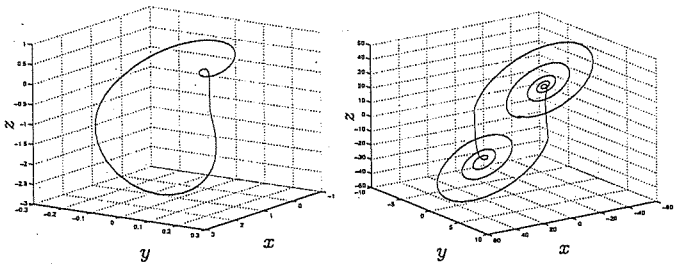
Here, we state that the homoclinic and heteroclinic Shil'nikov method are assuming the existence of a homoclinic and heteroclinic orbit in the dynamics of the system (3.6), respectively.

Theorem 3.1 (Homoclinic Shil'nikov Method) [236] *Given a third-order autonomous system in (3.6), where $\mathcal{Q}$ is a C^2 vector field on $\mathbb{R}^3$. Let x_e be an equilibrium point for (3.6). Suppose the following:*

- *The equilibrium point is a saddle focus whose eigenvalues satisfy the Shil'nikov inequality, that is*

$$0 < |\sigma| < |\gamma|.$$

- *There exists a homoclinic orbit $\mathcal{H}$ based at x_e.*



(a) (b)

Figure 3.3: Numerical simulation of (a) a homoclinic orbit and (b) a heteroclinic orbit in the dynamics of Chua's circuit.

Then,

- *The Shil'nikov map defined in a neighborhood of $\mathcal{H}$ possesses a countable number of Smale horseshoes in its discrete dynamics.*
- *For any sufficiently small C-perturbation ζ of $\mathcal{Q}$, the perturbed system*

$$\dot{x} = \zeta(x), \quad x \in \mathbb{R}^3$$

has at least a finite number of Smale horseshoes in the discrete dynamics of the Shil'nikov map defined near $\mathcal{H}$.

- *Both the original system (3.6) and the perturbed one exhibit horseshoe chaos (also known as homoclinic chaos).*

Homoclinic chaos in Chua's circuit

Chua's circuit is a member of the piecewise-linear vector field family $\mathcal{F}$ which is defined in [235]. The presence of homoclinic chaos in this field is demonstrated in [43] with a detailed bifurcation analysis. The vector field $\mathcal{F}$ possesses the following properties 1) piecewise-linear (and hence piecewise C^∞) consisting of three regions with two parallel planes as the boundary surfaces, and, 2) is odd symmetric with respect to the origin. The piecewise-linear nature of the vector field makes it possible to find the existence of a pair of homoclinic orbits $\mathcal{H}^{\mp}$ based at the origin. How to find homoclinic and also heteroclinic orbits in Chua's circuit has been discussed in [177] and [152]. Numerical simulation of a

homoclinic orbit in the dynamic of Chua's circuit has been in [236, 143]. Figure 3.3(a) shows the homoclinic orbit of Chua's circuit following parameter values in (3.3):

$$\alpha = 11.6047522, \beta = 15, m_0 = -1.142857143, m_1 =$$

and with initial condition

$$x(0) = 0.0086691, y(0) = 0.000958759, z($$

With the parameters of (3.7), one can find the gin are

$$\gamma = 2.9413, \quad \text{and} \quad \sigma \mp jw = -1.141$$

thus satisfying the Shil'nikov inequality. Hence the existence of Chua's circuit is proven.

Theorem 3.2 (Heteroclinic Shil'nikov Method) [236] *Given a third-order autonomous system in (3.6), where $\mathcal{Q}$ is a C^2 vector field on $\mathbb{R}^3$. Let x_{e_1} and x_{e_2} be two distinct equilibrium points for (3.6). Suppose the following:*

- *Both x_{e1} and x_{e2} are saddle foci that satisfy the Shil'nikov inequality*

$$0 < |\sigma_i| < |\gamma_i| \quad (i = 1, 2)$$

with the additional constraint

$$\sigma_1\sigma_2 > 0 \ \text{ or } \ \gamma_1\gamma_2 > 0.$$

- *There is a heteroclinic loop $\mathcal{H}_l$ joining x_{e1} to x_{e2} that is consists of two heteroclinic orbits $\mathcal{H}_i$ $(i = 1, 2)$.*

Then, the conclusions of Theorem 3.1 hold again, with $\mathcal{H}$ replaced by $\mathcal{H}_l$, the equilibrium point x_e by x_{e_i} $(i = 1, 2)$, and the homoclinic chaos by the corresponding term heteroclinic chaos.

Heteroclinic chaos in Chua's circuit

The presentation of heteroclinic orbit $\mathcal{H}_l$ in Figure 3.3(b) was found for Chua's circuit (3.3) with the following parameter values:

$$\alpha = -4.50746268737, \ \beta = 3.3373353844 \ m_0 = -2.4924 \ m_1 = -0.93$$

and the following initial condition

$$x(0) = 22.2861576, \ y(0) = 0.009506608, \ z(0) = -22.28662665$$

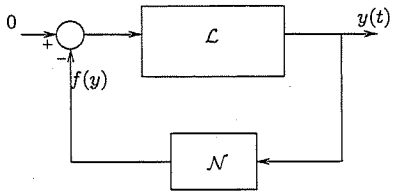


Figure 3.4: Lur'e system.

[236]. The heteroclinic orbit $\mathcal{H}_l$ connects the two saddle foci where $[x \ y \ z] = [\mp 22.32 \ 0 \ \pm 22.32]$ whose eigenvalues are

$$\gamma = -0.95006607987, \ \sigma \pm jw = 0.1330915934 \pm j1.044002616.$$

These eigenvalues satisfy the Shil'nikov inequality in Theorem 3.2. Hence the existence of heteroclinic chaos in Chua's circuit is proven [235].

3.1.1.2 Harmonic balance method

In 1992, Genesio and Tesi [91] proposed a new heuristic approach for the approximate determination of the existence and location of chaotic behavior of dynamical systems which can be represented in Lur'e form (Figure 3.4). The approach is based on the harmonic balance method which is well-known in the area of system and control for detecting nonlinear oscillations by means of the describing function method. The dynamic systems considered by Genesio and Tesi are of the form

$$q(D)y(t) + p(D)f(y) = 0 \tag{3.8}$$

where $y(t)$ is the output variable, D the differential operator, $p(\cdot)$ and $q(\cdot)$ are two polynomial operators and $f(\cdot)$ is a nonlinear real valued function. The system described by Eq. (3.8) can be considered as a feedback structure consisting of a linear part $\mathcal{L}$ and a nonlinear part $\mathcal{N}$ connected by two scalar variables y and $f(\cdot)$ as in Figure 3.4 The linear part $\mathcal{L}$ can be modeled by the transfer function (s is the complex variable)

$$L(s) = \frac{p(s)}{q(s)}. \tag{3.9}$$

Genesio and Tesi [89] formulated the following conjecture using a basic observation is that a chaotic motion is a kind of limit cycle where the periodic solution is perturbed by the existence of another characteristic of the system.

Conjecture 3.1 (Genesio and Tesi) [89] *A Lur'e feedback system presents chaotic behavior when a stable predicted limit cycle and an unstable equilibrium point interact with a suitable filtering effect along the system.*

The essential elements of the Conjecture 3.1 are the following;

Equilibrium Points (EPs): The constant output values y_e corresponding to the equilibrium solutions of system (3.8) which satisfy

$$y_e + L(0)f(y_e) = 0. \tag{3.10}$$

The stability of the EPs can be studied by linearizing $f(y)$ at y_e.

Predicted Limit Cycles (PLCs): In order to predict a limit cycle, it is assumed that the system (3.8) allows a solution of the form

$$y(t) = a_0 + a_1 \sin(\omega t). \tag{3.11}$$

Assume that the corresponding nonlinear output $f(y)$ is expanded in Fourier series as

$$f(y(t)) = N_0(a_0, a_1)a_0 + N_1(a_0, a_1)a_1 \sin(\omega t) + \cdots. \tag{3.12}$$

The nonlinear system $\mathcal{N}$ is characterized by the well-known describing function terms [14]

$$N_0(a_0, a_1) = \frac{1}{2\pi a_0} \int_{-\pi}^{\pi} f(a_0 + a_1 \sin(\phi))d\phi \tag{3.13}$$

$$N_1(a_0, a_1) = \frac{1}{\pi a_1} \int_{-\pi}^{\pi} f(a_0 + a_1 \sin(\phi)) \sin(\phi)d\phi. \tag{3.14}$$

By discarding the harmonics higher than one of Eq. (3.12), substituting Eqs. (3.11), (3.13) and (3.14) in Eq. (3.8) leads to the relations

$$a_0(1 + L(0)N_0(a_0, a_1)) = 0 \tag{3.15}$$

and

$$1 + L(jw)N_1(a_0, a_1) = 0 \tag{3.16}$$

which are the conditions to be satisfied by a PLC. Furthermore, these conditions constitute the harmonic balance method for the system (3.8). The stability of the predicted limit cycle can be evaluated from Eqs. (3.15) and (3.16) by means of the Loeb criterion [50].

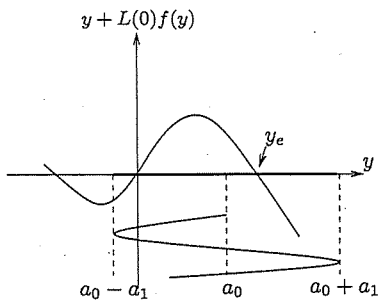


Figure 3.5: Interaction between predicted limit cycle (PLC) and equilibrium point (y_e).

Interaction: The condition expressing the possibility of interaction between PLC and EP follows from the solutions of Eqs. (3.15), (3.16) and (3.11) in the form

$$|y_e - a_0| \leq a_1. \tag{3.17}$$

Figure 3.5 shows the interaction between the PLC and EP (the EP y_e falls within the projection onto the $y-$axis of the PLC).

At this point, according the Conjecture 3.1, all the quantitative conditions have been specified. The last condition is the filtering effect which is the quantitative condition in Conjecture 3.1.

Filtering effect: The accuracy of the prediction of limit cycles for the system (3.8) is based on the filtering effect which was approximately translated in the inequality

$$|L(j\omega)| \gg |L(jk\omega)|, \quad k = 2, 3, \cdots \tag{3.18}$$

in [89]. The condition requires a suitable filtering effect of the system in the sense that high value of this element ensure a pure oscillation corresponding to the PLC whereas its reduction leads to a domain of parameters of chaotic behavior.

Chaos prediction on Chua's circuit using harmonic balance method

Chua's circuit (3.3) can be transformed into the following third-order scalar differential equation

$$\frac{\partial^3 x}{\partial \tau^3}+(1+\alpha)\frac{\partial^2 x}{\partial \tau^2}+\beta\frac{\partial x}{\partial \tau}+\beta\alpha x+\alpha\left(\frac{\partial^2 f(x)}{\partial \tau^2}+\frac{\partial f(x)}{\partial \tau}+\beta f(x)\right)=0. \quad (3.19)$$

The associated linear subsystem of Chua's circuit is described by the following transfer function

$$L(s)=\frac{p(s)}{q(s)}=\frac{\alpha(s^2+s+\beta)}{s^2+(1+\alpha)s^2+\beta s+\alpha\beta}. \quad (3.20)$$

The nonlinear subsystem here is represented by Eq. (3.5) with $k_0=-\frac{8}{7}$, $k_1=\frac{4}{63}$. In [90], the nonlinear subsystem is represented by the piecewise-linear function (3.4). The system has three equilibrium points which correspond to the solutions of equation (3.10). Since $L(0)=1$, three solutions of Eq. (3.10) are given by $x_{e_1}^{+}=-\frac{3}{2}$, $x_{e_0}=0$, $x_{e_1}^{-}=\frac{3}{2}$. In order to find the periodic solutions for Chua's circuit, consider the possible solution of Eq. (3.11). Since $L(j\omega)$ is a complex number, Eqs. (3.15) and (3.16) constitute three independent equations and can be solved with respect to a_0, a_1 and ω to give the parameters that define the approximate periodic solutions in the form Eq. (3.11). Since N_1 is a real number, we can obtain

$$\mathrm{Im}[L(j\omega)]=0 \quad (3.21)$$

from the Eq. (3.16). Hence, the oscillation frequency can be determined from this condition. Substituting Eq. (3.20) into Eq. (3.21) and solving for ω, we obtain

$$\omega^2=\beta-\left(\frac{1+\alpha}{2}\right)\mp\left[\left(\frac{1+\alpha}{2}\right)^2-\beta\right]^2. \quad (3.22)$$

Equation (3.22) gives us two real values ω' and ω'' if

$$\left(\frac{1+\alpha}{2}\right)^2>\beta>\alpha>1. \quad (3.23)$$

Formulas (3.13) and (3.14) give the describing functions terms of the nonlinearity defined by Eq. (3.5) as

$$N_0(a_0,a_1)=-\frac{8}{7}+\frac{4}{63}a_0^2+\frac{2}{21}a_1^2 \quad (3.24)$$

$$N_1(a_0,a_1)=-\frac{8}{7}+\frac{4}{21}a_0^2+\frac{1}{21}a_1^2. \quad (3.25)$$

The application of harmonic balance with Eqs. (3.15) and (3.16) yields the parameters

$$\widehat{a}_0=\mp\sqrt{\frac{63}{20}\left(\frac{15}{7}-\frac{2}{\widehat{L}}\right)},\quad \widehat{a}_1=\sqrt{\frac{21}{5}\left(-\frac{5}{7}+\frac{1}{\widehat{L}}\right)} \quad (3.26)$$

where

$$\widehat{L}=L(j\omega')=\frac{2\alpha}{(1+\alpha)+\sqrt{(1+\alpha)^2-4\beta}}. \quad (3.27)$$

Equation (3.26) requires in addition to Eq. (3.23) the following PLC existence condition $\widehat{L}<\frac{7}{5}$ for $\alpha>7/3$. Observe that Eq. (3.26) indicates that two symmetrical PLCs with respect to the origin and they have projections on x around the EPs $\mp 3/2$. Therefore, their interaction must be considered toward the EP at x_{e_0}. The linearization of the nonlinearity (3.5) at x_{e_0} indicates an unstable eigenvalue for the parameter region of interest ($\alpha>7/3$). Also stability of the PLCs can be checked by the application of the Loeb criterion [50] which leads to the formula

$$\left.\frac{\partial N_1(a_0(a_1),a_1)}{\partial a_1}\right|_{a_1=\widehat{a}_1}=-\frac{15\widehat{a}_1}{2}<0$$

which indicates that these periodic motions are stable. So far we have shown that Chua's circuit has two stable PLCs and an unstable EP at the origin which are the main ingredients of the method. The condition (3.5) of interaction between PLCs and EP can be written as $a_1\geq|a_0|$ and term of (3.26) results into $\widehat{L}<\frac{14}{13}$. Using the expression (3.27), the relation becomes

$$\beta<\frac{13}{196}\alpha^2+\frac{13}{14}\alpha \quad (3.28)$$

which is the border of the chaotic region in the (α,β)-plane. This is found to be good in agreement with the bifurcation diagram in [43] and [131]. The filtering effect is evaluated by the distortion index Δ [92, 90] and can be expressed by the quantity

$$\Delta=\frac{\|\widehat{x}(t)-x(t)\|_2}{\|x(t)\|_2}$$

where $\widehat{x}(t)$ denotes the steady state periodic output of the open loop in response to $x(t)$ and $\|.\|_2$ is the L_2 norm on the period $2\pi/\omega$. For Chua's circuit, the value of the distortion index is given for $0.03<\Delta<0.06$ in [92].

3.1.2 Realization of Chua's circuit

Since all of the linear elements (capacitor, resistor and inductor) are readily available as two-terminal devices, the principal concern of the realization is to realize the Chua's diode (N_R) with active and passive elements. Several implementations of Chua's diode exist in the literature (including diodes [174], operational amplifiers [323], transistors [176], operational transconductance amplifiers [57] and current feedback operational amplifiers [230, 78] to

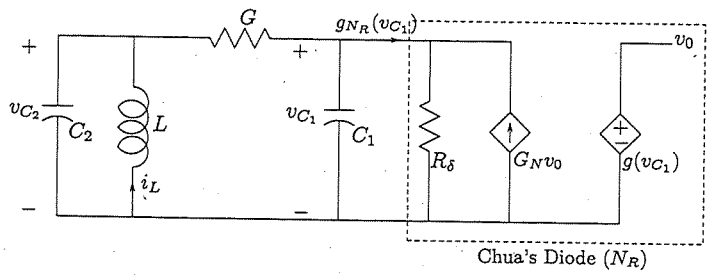


Figure 3.6: A circuit model for Chua's circuit which is formed to be a cell in a CNN [100] ($G_N = 1S$).

synthesize the Chua's diode). A systematic procedure for synthesizing Chua's diode with independently adjustable slopes and breakpoints was described by Kennedy [130]. An entire monolithic Chua's circuit has been fabricated by Delgado-Restituto and Rodriguez-Vasquez [68]. Furthermore, Arena *et al.* [5] have proved that Chua's circuit can be realized by three SC-CNN (see Section 2.2.2) cells, which will be explained in Section 3.2.1.

Here, we consider to realize an uncoupled cell of a generalized CNN based on Chua's circuit which was introduced in Section 2.2.5.3. The cell (see Figure 3.6) is obtained from Chua's circuit by decomposing Chua's diode into a linear positive resistor (R_δ), a nonlinear voltage controlled voltage source ($v_0 = g(v_{C_1})$) and a linear voltage controlled current sources ($i = G_N v_0$ and $G_N = 1S$). For this type of realization the corresponding state equations of Chua's circuit are given by

$$\begin{cases} \dot{x} &= \alpha[-(1+\delta)x + y + g(x)] \\ \dot{y} &= x - y + z \\ \dot{z} &= -\beta y \end{cases} \tag{3.29}$$

with $\delta = 1$ and

$$g(x) = \overline{m}_1 x + \frac{1}{2}(\overline{m}_0 - \overline{m}_1)(|x + E_1| - |x - E_1|) \tag{3.30}$$

with $f(x) = -g(x) + (1+\delta)x$ for $E_1 = 1$.

The realized circuit has been shown in Figure 3.7 where bifurcation parameter is chosen as R_δ [298, 296]. With this implementation the slopes and breakpoints of the characteristic can be independently adjusted as in [130]. The slopes and breakpoints are given by

$$\overline{m}_0 = \frac{R_{21}}{R_{11}}\frac{R_4}{R_4 + R_3} + \frac{R_{22}}{R_{12}},\ \overline{m}_1 = \frac{R_{22}}{R_{12}} \text{ and } E_1 = \frac{R_{11}}{R_{21}}E_{sat}.$$

By setting $C_1 = 100nF$, $C_2 = 1nF$, $L = 68mH$, $G = 1/10k\Omega, \overline{m}_0 = \frac{15}{7}, \overline{m}_1 = \frac{12}{7}$, $E_1 = 1V$, $R_1 = R_{11} = R_{12} = 10k\Omega$, $R_2 = 100k\Omega$, $R_3 = 29k\Omega$, $R_4 = 1k\Omega$, $R_5 = 100k\Omega$, $R_{22} = 17k\Omega$, $R_{21} = 120k\Omega$ and adjusting the value of R_δ in the circuit of Figure 3.7, we have observed periodic orbits with period T in the (v_{C_1}, v_{C_2})-plane for the resistor R_δ in the range of 7870Ω to 8217Ω, periodic orbit with period $2T$ for the range of 8218Ω to 8277Ω, periodic orbit with period $4T$ for the range of 8278Ω to 8291Ω, double scroll orbit for the range of 8400Ω to 10400Ω. In Figure 3.8 periodic orbit with period T, with period $2T$ and with period $4T$, double scroll orbit and their spectra have been given. The observed power spectra confirm period doubling route to chaos [238, 29]. The global bifurcation landscape of Chua's circuit is obtained by Komuro *et al.* [152] and the bifurcation analysis of Chua's circuit with the smooth nonlinearity (3.5) is presented in [135].

3.2 Generalized Chua's circuit

Among the generalizations that have been proposed to the Chua's circuit one basically has two categories: either one modifies the nonlinear characteristic or one tries to increase the dimensionality of the circuit [140, 95, 287, 166]. Originally, by Suykens and Vandewalle [245, 246] a family of so-called n-double scroll attractors was introduced, where n denotes a natural number. This was done by considering additional breakpoints in the nonlinear characteristic. A more complete family of n-scroll instead of n-double scroll attractors, which also allows for an odd instead of an even number of scrolls, has been obtained from the generalized Chua's circuit proposed by Suykens, Huang and Chua in [244].

The following generalized Chua's circuit has been proposed in [244] by Suykens, Huang and Chua:

$$\begin{cases} \dot{x} &= \alpha\,[y - h(x)] \\ \dot{y} &= x - y + z \\ \dot{z} &= -\beta y \end{cases} \tag{3.31}$$

with piecewise linear characteristic

$$h(x) = m_{2q-1}x + \frac{1}{2}\sum_{i=1}^{2q-1}(m_{i-1} - m_i)\,(|x + c_i| - |x - c_i|) \tag{3.32}$$

which consists of multiple breakpoints, where q denotes a natural number. The generalized Chua's circuit (3.31) is decribed by three scalar and two vector valued parameters $\{\alpha,\ \beta,\ q,\ m,\ c\}$ where $m = [m_0\ m_1 \cdots m_{2q-1}]$,

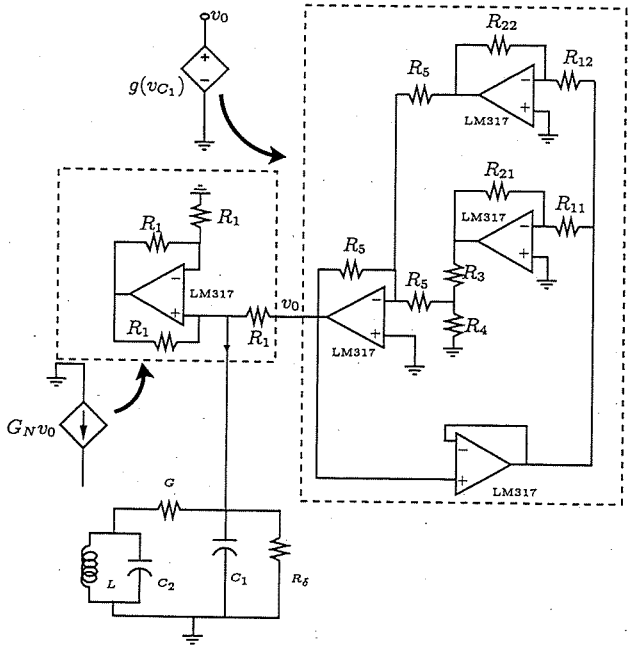


Figure 3.7: Chua's circuit realized by operational amplifiers. The realizations of the linear voltage controlled current source ($G_N v_0$) and nonlinear voltage controlled voltage source ($g(v_0)$) of the circuit model (Figure 3.6) are indicated.

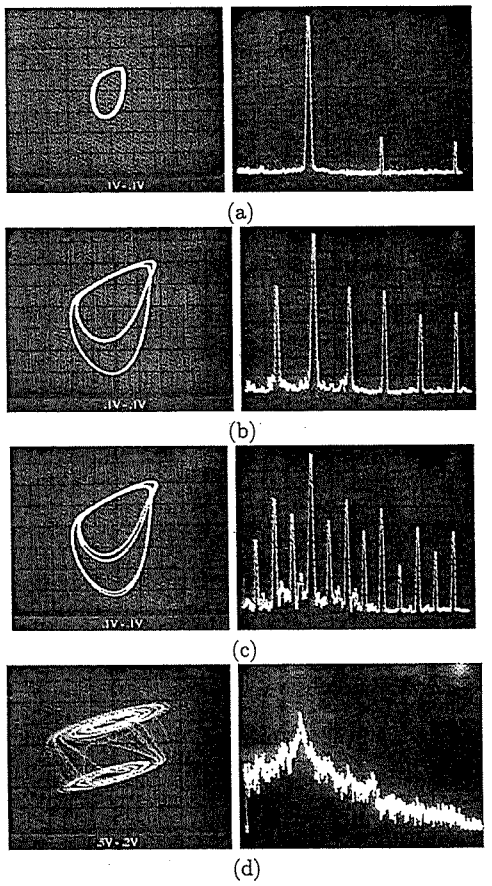


Figure 3.8: Periodic orbit with period (a, Left) T (b, Left) $2T$ (c, Left) $4T$ and (d, Left) double scroll attractor observed with the circuit of Figure 3.7 in the (v_{C_1}, v_{C_2})-plane. Horizontal scale $0.1V/div$, vertical scale $0.1V/div$. (a, Right)-(d, Right) Power spectra of the capacitor voltage v_{C_1} on $(P_{v_{C_1}}(\omega), \omega)$ for Period T, $2T$, $4T$ and double scroll attractor, respectively.

$c = [c_1\ c_2 \cdots c_{2q-1}]$. A double scroll attractor is obtained from the generalized Chua's circuit for the parameters {9, 14.28, 1, $[-1/7\ +2/7]$, 1} which is equal to the Chua's equation (3.3) for the parameters $\{9, 14.28, -8/7, -5/7\}$.

Furthermore, Suykens and Vandewalle [245] proposed an n-double scroll generation scheme based on an alternative heuristic method for studying nonlinear differential equations which is quasilinear approach. The idea is simply to keep the eigenvalue configurations for the new equilibrium points in relation to the additional breakpoints. The structure of the eigenvalue configurations are given by Suykens, Huang and Chua [244] in order to obtain n-scroll attractors. Although this method is heuristic, it gives some global insight into the state-space behavior of the system. Furthermore, harmonic balance method has been applied to the generalized Chua's circuit by Savacı *et al.* [224] and the existence and location of n-scroll attractors have been verified.

One obtains a n-double scroll attractor taking the parameters {9, 14.28, q, m, c} where

2-double scroll (4-scroll) [245, 246]

$$\begin{aligned} q &= 2 \\ m &= [-1/7\ +2/7\ -4/7\ +2/7] \\ c &= [1\ 2.15\ 3.6] \end{aligned} \tag{3.33}$$

3-double scroll (6-scroll) [245, 246]

$$\begin{aligned} q &= 3 \\ m &= [-1/7\ +2/7\ -4/7\ +2/7\ -4/7\ +2/7] \\ c &= [1\ 2.15\ 3,6\ 8.2\ 13]. \end{aligned} \tag{3.34}$$

Figure 3.9(a) and 3.9(b) shows these two attractors. The components of the vector m have alternating signs. The sign of m_0 is negative in the case of n-double scrolls (even number of scrolls). n-Scroll attractors with an odd number of scrolls are generated from Eqs. (3.31)-(3.32) by taking the same values for the α, β parameters but opposite sign for the vector m in the nonlinearity where

1-scroll [244]

$$\begin{aligned} q &= 1 \\ m &= [+1/7\ -2/7] \end{aligned} \tag{3.35}$$

$$c = 1$$

3-scroll [244]

$$\begin{aligned} q &= 2 \\ m &= [+0.9/7\ -3/7\ +3.5/7\ -2.4/7] \\ c &= [1\ 2.15\ 4] \end{aligned} \tag{3.36}$$

5-scroll [244]

$$\begin{aligned} q &= 3 \\ m &= [+0.9/7\ -3/7\ +3.5/7\ -2.7/7\ +4/7\ -2.4/7] \\ c &= [1\ 2.15\ 3.6\ 6.2\ 9] \end{aligned} \tag{3.37}$$

7-scroll [244]

$$\begin{aligned} q &= 3 \\ m &= [+0.9/7\ -3/7\ +3.5/7\ -2.4/7 \\ &\quad +2.52/7\ -1.68/7\ +2.52/7\ -1.68/7] \\ c &= [1\ 2.15\ 3.6\ 6.2\ 9\ 14\ 25]. \end{aligned} \tag{3.38}$$

3- and 5-scroll attractors are shown in Figure 3.9(c) and Figure 3.9(d), respectively.

3.2.1 Realization of n-scroll attractors from generalized Chua's circuits

In 1995, Arena *et al.* [5] introduced a method to realize Chua's circuit form a connection of three space variant 1-D SC-CNN cells given by

$$\dot{x}_j = -x_j + a_j y_j + \sum_{i=1}^{3} \hat{a}_{j,i} x_{j-2+i} + \sum_{i=1}^{3} a_{j,i} y_{j-2+i} + I_j \tag{3.39}$$

with j being the cell index, x_j the state variable and y_j the cell output with

$$y_j = \frac{1}{2}(|x_j + 1| - |x_j - 1|), \tag{3.40}$$

a_j a constant parameter and I_j a threshold. $\hat{a}_{j,i}$ and $a_{j,i}$ are weights from the combinations of the outputs and state variables of the connected cells, respectively. Arena *et al.* [5] experimentally showed that a double scroll attractor

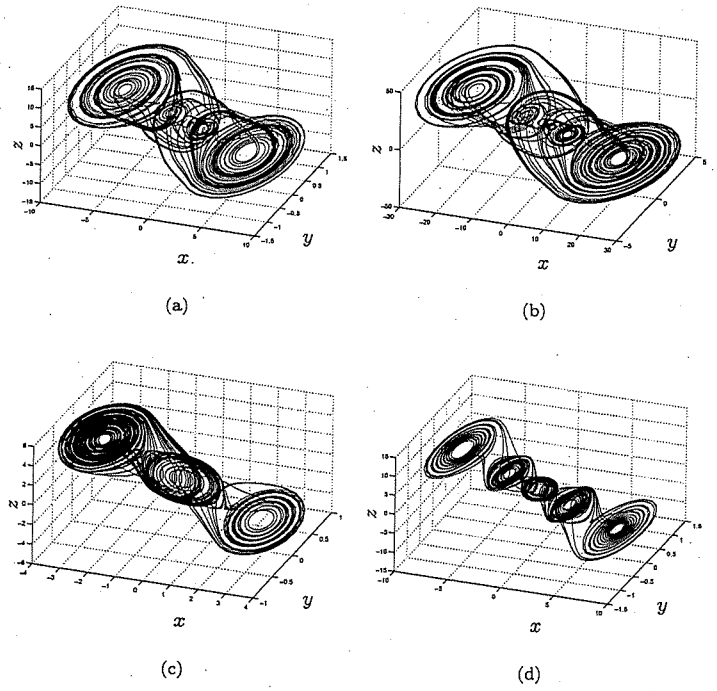


Figure 3.9: Phase portraits of a (a) 2-double scroll (4-scroll), (b) 3-double scroll (6-scroll), (c) 3-scroll, (d) 5-scroll attractors obtained from a generalized Chua's circuit.

can be obtained by three state controlled CNN cells with the parameters

$$\begin{aligned} &a_{i,j} = 0 \ (i,j = 1,2,3), \ \hat{a}_{1,3} = \hat{a}_{2,2} = \hat{a}_{3,1} = 0, \ a_2 = 0, \\ &a_1 = \alpha(m_1 - m_0), \ \hat{a}_{1,1} = 1 - \alpha m_1, \ \hat{a}_{1,2} = \alpha, \\ &\hat{a}_{2,1} = \hat{a}_{3,3} = \hat{a}_{2,3} = 1, \ \hat{a}_{3,2} = -\beta. \end{aligned} \tag{3.41}$$

In fact, the SC-CNN (Eq. (3.39)) with the parameters (3.41) is equal the Chua's equation (3.3) with the parameters $\{\alpha, \beta, m_0, m_1\}$.

Then the first experimental confirmation of a n-double scroll attractor has been given by Arena *et al.* [7, 6] with the same methodology. The generalized Chua's circuit has been derived from a connection of three SC-CNN cells [5] with a further generalization that is done on the output nonlinearity. The generalized nonlinear output is given by

$$y = \frac{1}{2}\sum_{k=1}^{2q-1}(m_{k-1} - m_k)(|x + c_k| - |x - c_k|) \tag{3.42}$$

where c_k are the breakpoints and the coefficients m_k are the slopes of the segments. The SC-CNN (3.39) with the parameters (3.41) but $a_1 = \alpha$ and $s_{1,1} = 1 - \alpha m_{2q-1}$ is equal the generalized Chua's circuit (3.31) with the parameters $\{\alpha, \beta, q, m, c\}$. A 2-double scroll attractor has been obtained from the circuit implementation of the three SC-CNN cells for the parameters $\alpha = 9$, $\beta = 14, 28$, m and c were given in Eq. (3.33).

3.2.1.1 Realization of 3- and 5-scroll attractors

In order to realize n-scroll attractors, we generalized the nonlinearity of the system (3.29) which is given by

$$g(x) = (1 + \delta)x - h(x) \tag{3.43}$$

where $\delta = 1$ and $h(x)$ is the nonlinearity of the generalized Chua's circuit (3.32). Hence, the slopes of the each segment of $g(x)$ is found by

$$\overline{m}_i = (1 + \delta) - m_i, \quad i = 1, 2, ..., 2q - 1 \tag{3.44}$$

where m_i are the slopes of $h(x)$ (3.32). We used the same circuit model which is given in Figure 3.6 (the corresponding state equations are given in Eq. (3.29)). The nonlinearity is synthesized by a nonlinear VCVS which is realized by operational amplifiers in such a way that this realization allows that the slopes and breakpoints of the nonlinearity can be independently adjusted [302].

For 5-scroll attractors, $q = 3$ and $g(x)$ is obtained by a sum of 7 terms which are separately realized by inverting/noninverting amplifiers and voltage branches

(see Figure 3.10). First, 2 inverting amplifiers of seven terms realize slopes of the last PWL part ([9 V_{cc}] and [$-V_{cc}$ -9]). The outputs of the other terms are saturated when their input reaches the desired breakpoint. The breakpoints are adjusted with gains of inverting/noninverting amplifiers:

$$c_5 = \frac{E_{sat}}{1+\frac{R_{31}}{R_{32}}},\ c_4 = \frac{E_{sat}}{\frac{R_{51}}{R_{52}}},\ c_3 = \frac{E_{sat}}{1+\frac{R_{71}}{R_{72}}},\ c_2 = \frac{E_{sat}}{\frac{R_{92}}{R_{91}}},\ c_1 = \frac{E_{sat}}{1+\frac{R_{111}}{R_{110}}}. \quad (3.45)$$

The slopes are adjusted with the gains of inverting/noninverting amplifiers and voltage branches:

$$\begin{aligned}
\overline{m}_5 &= -\tfrac{R_{12}}{R_{11}} - \tfrac{R_{22}}{R_{21}} \\
\overline{m}_4 &= \overline{m}_5 + \left(1+\tfrac{R_{31}}{R_{32}}\right)\tfrac{R_{42}}{R_{42}+R_{41}} \\
\overline{m}_3 &= \overline{m}_4 - \tfrac{R_{51}}{R_{52}}\tfrac{R_{62}}{R_{62}+R_{61}} \\
\overline{m}_2 &= \overline{m}_3 + \left(1+\tfrac{R_{71}}{R_{72}}\right)\tfrac{R_{82}}{R_{82}+R_{81}} \\
\overline{m}_1 &= \overline{m}_2 - \tfrac{R_{92}}{R_{91}}\tfrac{R_{101}}{R_{101}+R_{100}} \\
\overline{m}_0 &= \overline{m}_1 + \left(1+\tfrac{R_{111}}{R_{110}}\right)\tfrac{R_{121}}{R_{121}+R_{120}}.
\end{aligned} \quad (3.46)$$

In our realization of the PWL characteristic, all slopes and breakpoints are adjusted independently. The corresponding capacitor and resistor values for the 5-scroll attractor Eq. (3.31-3.32) with the parameters $\alpha = 9$, $\beta = 14.28$, m and c where are in Eq. (3.37), are given by $C_1 = 1/9F$, $C_2 = 1F$, $R = 1\Omega$, $R_\delta = 1\Omega$, $L = 0.07H$. When current and time are rescaled by a factor 10000, the parameters become $C_1 = 10/9nF$, $C_2 = 10nF$, $R = 10k\Omega$, $R_\delta = 10k\Omega$ and $L = 70mH$. This results into

$$\overline{m} = [0.18714mS\ \ 0.24286mS\ \ 0.15mS\ \ 0.23857mS\ \ 0.14286mS\ \ 0.234229mS]$$

with unchanged location of the breakpoints. C_1 and L are not standard values, therefore we have chosen $C_1 = 1nF$ and $L = 70.2mH$. The implemented circuit and $g(v_{C1})$ characteristic of the VCVS are shown in Figure 3.10 and Figure 3.11, respectively. The component list is given in Table 3.1. In our experiments, for $C_2 = 10nF$ the circuit was unstable such that we changed it into $20nF$, corresponding to doubling the values for α and β. In the computer simulations a 5-scroll attractor is also obtained for these α and β values. By adjusting the value of R in Figure 3.10 from 8100Ω to 9200Ω the generalized Chua's circuit goes from a 1-scroll to a 3-scroll and the 5-scroll, as illustrated in Figure 3.12. A two dimensional projection of the attractor is obtained by connecting v_{C1} and v_{C2} to the X and Y channels respectively of an $X-Y$ oscilloscope. Figure 3.13 shows the spectrum of 3- and 5-scroll attractors for v_{C1}.

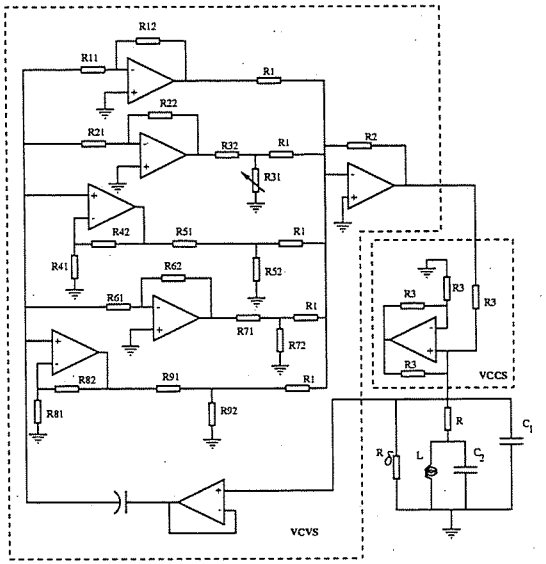


Figure 3.10: Realization of the 5-scroll circuit (generalized Chua's circuit).

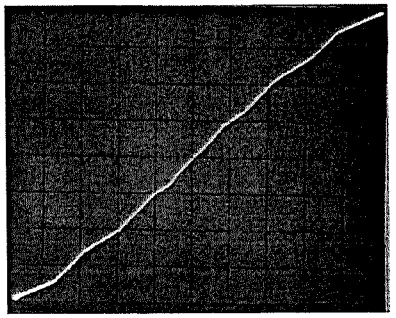
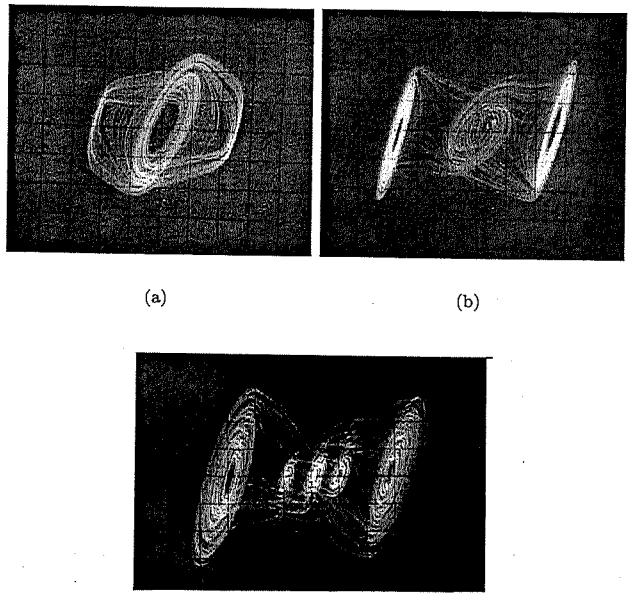


Figure 3.11: The $g(v_{C1})$ characteristic of the VCVS (horizontal axis v_{C1} $2V/div$, vertical axis $g(v_{C1})$ $2V/div$).



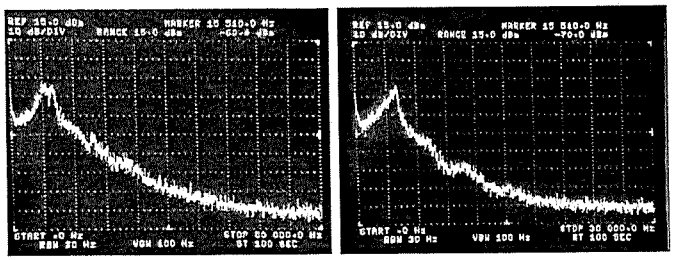
(a) (b)

(c)

Figure 3.12: Experimental confirmation of 1-scroll, 3-scroll and 5-scroll attractors: (a) 1-scroll $R = 8100\,\Omega$; (b) 3-scroll $R = 12280\Omega$ (horizontal axis v_{C1} $0.5V/div$, vertical axis v_{C2} $0.1V/div$); (c) 5-scroll $R = 12740\,\Omega$ (horizontal axis v_{C1} $1V/div$, vertical axis v_{C2} $0.2V/div$).

Element	Description	Value	Tolerance
	Op amp (TL082)		
R11	$\frac{1}{4}$W Resistor	$13k\Omega$	$\mp$%1
R12	$\frac{1}{4}$W Resistor	$11k\Omega$	$\mp$%1
R21	$\frac{1}{4}$W Resistor	$5.1k\Omega$	$\mp$%1
R22	$\frac{1}{4}$W Resistor	$24k\Omega$	$\mp$%1
R31	$\frac{1}{4}$W Resistor	$1.3k\Omega$	$\mp$%1
R32	$\frac{1}{4}$W Resistor	$13k\Omega$	$\mp$%1
R41	$\frac{1}{4}$W Resistor	$1.3k\Omega$	$\mp$%1
R42	$\frac{1}{4}$W Resistor	$8.2k\Omega$	$\mp$%1
R51	$\frac{1}{4}$W Resistor	$24k\Omega$	$\mp$%1
R52	$\frac{1}{4}$W Resistor	$1.5k\Omega$	$\mp$%1
R61	$\frac{1}{4}$W Resistor	$1.2k\Omega$	$\mp$%1
R62	$\frac{1}{4}$W Resistor	$20k\Omega$	$\mp$%1
R71	$\frac{1}{4}$W Resistor	$68k\Omega$	$\mp$%1
R72	$\frac{1}{4}$W Resistor	$1.8k\Omega$	$\mp$%1
R81	$\frac{1}{4}$W Resistor	$1.1k\Omega$	$\mp$%1
R82	$\frac{1}{4}$W Resistor	$30k\Omega$	$\mp$%1
R91	$\frac{1}{4}$W Resistor	$130k\Omega + 11k\Omega$	$\mp$%1
R92	$\frac{1}{4}$W Resistor	$2.2k\Omega$	$\mp$%1
R1	$\frac{1}{4}$W Resistor	$100k\Omega$	$\mp$%1
R2	$\frac{1}{4}$W Resistor	$100k\Omega$	$\mp$%1
R3	$\frac{1}{4}$W Resistor	$5.1k\Omega$	$\mp$%1
R	$\frac{1}{4}$W Resistor	$10k\Omega$	$\mp$%1
R_6	$\frac{1}{4}$W Resistor	$10k\Omega$	$\mp$%1
C1	Capacitor	$1nF$	$\mp$%0.1
C2	Capacitor	$20nF$	$\mp$%0.1
L	Ferroxcube self	$68mH + 2.2mH$	

Table 3.1: Component list for Figure 3.10



(a) (b)

Figure 3.13: Power spectra measured from v_{C1}: (a) 3-scroll; (b) 5-scroll attractors.

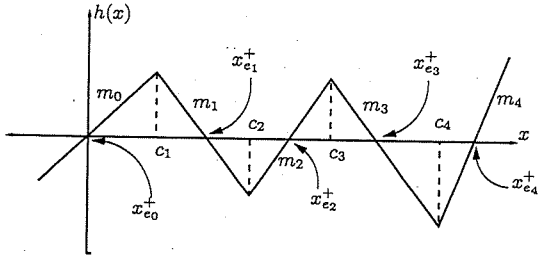


Figure 3.14: PWL nonlinearity of the generalized Chua's circuit for generating n-scroll attractors.

3.2.1.2 On the realization of n-scroll attractors

Until 1999, the largest number of n that has been experimentally confirmed was 5. Implementation of 6 scrolls and more needs a larger dynamic range, requiring higher voltage supply appropriate differential amplifiers or a convenient scaling of voltages, which was also remarked in [6]. In Section 3.2.1 and 3.2.1.1 the operational amplifiers are used to realize the sum of the non-linear terms which are separately realized by inverting and non-inverting amplifiers. In this type of realization, breakpoints are adjusted using the saturation voltage of the operational amplifiers. In order to realize 6-scrolls and more, the breakpoints exceed the operational amplifier's voltage supply. Therefore, we make an appropriate scaling of the nonlinearity by scaling the breakpoints of the PWL characteristic. On the other hand, the eigenvalue configurations of the Jacobians, evaluated at the equilibrium points, remain the same.

The equilibrium points of the generalized Chua's circuit (3.31-3.32) are given by

$$\begin{cases} h(x) &= 0 \\ x &= -z \\ y &= 0. \end{cases} \tag{3.47}$$

The equilibrium points can be derived from Figure 3.14. Equilibrium points are located at the intersection of the nonlinear function $h(x)$ with the x-axis. They are given by

$$m_0 c_1 + \sum_{j=1}^{l-1} m_j (c_{j+1} - c_j) + m_l (x_{e_l}^+ - c_l) = 0, \quad l = 1, ..., 2q-1 \tag{3.48}$$

and $x_{e_l}^- = -x_{e_l}^+$. The Jacobian matrix, evaluated at the first equilibrium point $\mathrm{eq}_0 = [0\ 0\ 0]^T$ and the i-th equilibrium point $\mathrm{eq}_i^{\pm} = [x_{e_i}^{\pm}\ y_{e_i}\ z_{e_i}]^T =$

$[x_{e_i}{}^{\pm}\ 0\ x_{e_i}{}^{\mp}]^T$, is equal to

$$J(\mathrm{eq}_0) = \begin{bmatrix} 0 & \alpha & 0 \\ 1 & -1 & 1 \\ 0 & -\beta & 0 \end{bmatrix} + \begin{bmatrix} -\alpha m_0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{3.49}$$

$$J(\mathrm{eq}_i) = \begin{bmatrix} 0 & \alpha & 0 \\ 1 & -1 & 1 \\ 0 & -\beta & 0 \end{bmatrix} + \begin{bmatrix} -\alpha m_i & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \ i = 1, ..., 2q-1. \tag{3.50}$$

Now, we scale the breakpoints of the piecewise linear characteristic with a factor k in such a way that all the slopes remain the same. After this scaling, the PWL characteristic becomes

$$h(x) = m_{2q-1} x + \frac{1}{2} \sum_{i=1}^{2q-1} (m_{i-1} - m_i)(|x + \frac{c_i}{k}| - |x - \frac{c_i}{k}|). \tag{3.51}$$

The new equilibrium points $[\frac{x_{e_l}{}^{\pm}}{k},\ 0,\ \frac{x_{e_l}{}^{\mp}}{k}]^T$ are scaled with the same factor k, which can be seen from

$$m_0 \frac{c_1}{k} + \sum_{j=1}^{l-1} m_j (\frac{c_{j+1}}{k} - \frac{c_j}{k}) + m_l (\frac{x_{e_l}}{k} - \frac{c_l}{k}) = 0, \quad l = 1, ..., 2q-1. \tag{3.52}$$

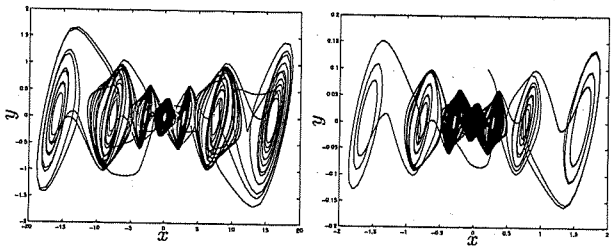
The eigenvalue configurations of the Jacobian matrix evaluated at this equilibrium points remain the same as Eqs. (3.49-3.50). Figure 3.15 shows computer simulation results for a 7-scroll attractor from generalized Chua's circuit with the parameters $\alpha = 9$, $\beta = 14.286$. m and c which are given in Eq. (3.38) for $k = 1$ and $k = 10$. The resulting circuit for $k = 10$ has a dynamic range which is feasible for implementation.

In order to realize a 6-scroll attractor, we scale the breakpoints of the PWL characteristic with $k = 4$. The realized circuit for $k = 4$ is given in Figure 3.16 with a component list in Table 3.2. The 6-scroll has been obtained for $\alpha = 18$, $\beta = 28.56$; scaled breakpoints $c = [0.25\ 0.53\ 0.9\ 2\ 3.25]$ and m are given in Eq. (3.34). A two dimensional projection of a 6-scroll attractor (see Figure 3.17) is obtained by connecting v_{C_1} and v_{C_2} to a $X - Y$ oscilloscope.

3.2.1.3 Alternative realizations of n-scroll attractors

In 2001, Tang *et al.* [262] demonstrated that n-scroll attractors can be obtained with a simple sine or cosine function. The approach provides a continuous function for generating n-scroll attractors and a guideline so that a different number of scrolls can be designed by modifying two variables in the introduced function. With the use of a commercial trigonometric function chip (AD639[1]), an

[1] Analog Devices [1990] "Linear products data book," *Norwood, Massachusetts, USA.*



(a) (b)

Figure 3.15: 7-Scroll attractor from a generalized Chua's Circuit in the (x, y)-plane: nonlinearities for (a) $k = 1$, (b) $k = 10$. Important to notice is also the different scaling of the Figures.

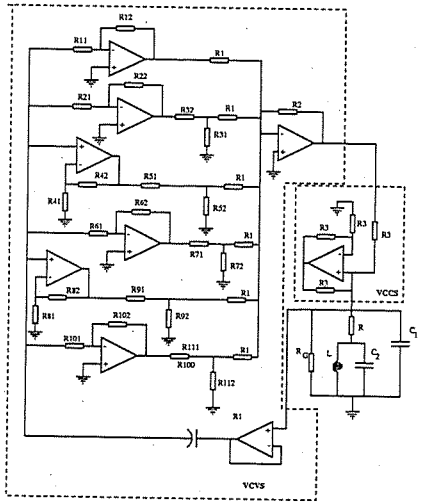


Figure 3.16: Realized circuit for 6-scroll attractor from a generalized Chua's circuit.

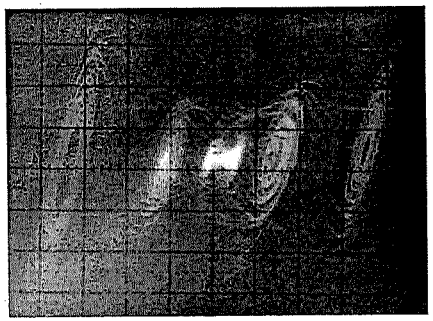


Figure 3.17: Experimental confirmation of 6-scroll attractor ($k = 4$) (horizontal axis v_{C_1} $(1V/div)$, vertical axis v_{C_2} $(0.2V/div)$).

Element	Description	Value	Tolerance
	Op amp (TL082)		
R11	¼W Resistor	13kΩ	∓%1
R12	¼W Resistor	11kΩ	∓%1
R21	¼W Resistor	5.1kΩ	∓%1
R22	¼W Resistor	24kΩ	∓%1
R31	¼W Resistor	1.3kΩ	∓%1
R32	¼W Resistor	13kΩ	∓%1
R41	¼W Resistor	1.3kΩ	∓%1
R42	¼W Resistor	8.2kΩ	∓%1
R51	¼W Resistor	24kΩ	∓%1
R52	¼W Resistor	1.5kΩ	∓%1
R61	¼W Resistor	1.2kΩ	∓%1
R62	¼W Resistor	20kΩ	∓%1
R71	¼W Resistor	68kΩ	∓%1
R72	¼W Resistor	1.8kΩ	∓%1
R81	¼W Resistor	1.1kΩ	∓%1
R82	¼W Resistor	30kΩ	∓%1
R91	¼W Resistor	130kΩ + 11IΩ	∓%1
R92	¼W Resistor	2.2kΩ	∓%1
R101	¼W Resistor	1kΩ	∓%1
R102	¼W Resistor	60kkΩ	∓%1
R111	¼W Resistor	300kΩ + 6.8kΩ	∓%1
R112	¼W Resistor	1.1kΩ	∓%1
R1	¼W Resistor	100kΩ	∓%1
R2	¼W Resistor	100kΩ	∓%1
R3	¼W Resistor	5.1kΩ	∓%1
R	¼W Resistor	10kΩ	∓%1
R_5	¼W Resistor	10kΩ	∓%1
C1	Capacitor	$1nF$	∓%0.1
C2	Capacitor	$20nF$	∓%0.1
L	Ferroxcube self	$68mH$ +$2.2mH$	

Table 3.2: Component list for the circuit in Figure 3.16.

electronic circuit was designed and implemented in [262, 261]. The introduced continuous function for generalized Chua's circuit (3.31) instead of the PWL function (3.32) is described by

$$h(x) = \begin{cases} \frac{b\pi}{2s}(x - 2ac), & \text{if } x \geq 2ac \\ -b\sin\left(\frac{x\pi}{2a} + d\right), & \text{if } -2ac < x < 2ac \\ \frac{b\pi}{2s}(x + 2ac), & \text{if } x \leq -2ac. \end{cases} \tag{3.53}$$

An n-scroll attractor is generated with the following relationship [262]:

$$n = c + 1 \tag{3.54}$$

and

$$d = \begin{cases} \pi, & \text{if } n \text{ is odd} \\ 0, & \text{if } n \text{ is even.} \end{cases} \tag{3.55}$$

2-scroll, 3-scroll, 4-scroll and 8-scroll attractors are generated from Eqs. (3.31) and (3.53) when $\alpha = 10.814$, $\beta = 14.0$, $a = 1.3$, $b = 0.11$ $c = 1, 2, 3$ and 7, respectively [262]. Furthermore, Tang *et al.* [262] reported the observation of a 7-scroll, 8-scroll and 9-scroll attractors on a circuit for the first time.

Recently, a systematic approach to realize multi segment piecewise linear functions has been developed by Zhong *et al.* [325] in order to generate multi-scroll from generalized Chua's circuit. They used negative resistors as building blocks where the slopes and breakpoints of its (v, i)-characteristic can be independently tuned. The attractors created by their design can have either an even or odd number of scrolls. Furthermore, they have obtained attractors with a large number of scrolls for a certain dynamic range of the opamps by tuning the slopes and breakpoints.

More recently, Fujiwara *et al.* [86] have implemented a prototype chip of generalized Chua's circuit with MOSIS TSMC 0.35μm CMOS process. PWL function (3.32) of generalized Chua's circuit is implemented by parallel combinations of N-type conductance circuits with floating-gate MOSFETs. A high-Q CMOS active inductor circuit is used as a simulated inductor. A 2-scroll and 3-scroll attractors have been experimentally observed on the prototype chip [86].

3.3 Families of scroll grid attractors

Here, we consider a new chaos generator which has a simple circuit implementation. The strange attractor families generated from the new chaos generators

are called *scroll grid attractors* [312, 310, 311, 252]. For these families it is possible to cover the whole state space with scrolls. The new attractor families are classified into three sub-families according to the location of the equilibrium points:

- *1-D grid scroll attractor family:* This corresponds with the original n-scroll attractors [244]. The equilibrium points of this family are located on a line and the scrolls generated from the generalized nonlinearity are located around that line along the x state variable direction in state space. Furthermore, the x state variable is also the variable on which the nonlinearity operates.
- *2-D grid scroll attractor family:* In this family, the system consists of two nonlinear functions operating on the x and y state variables. The equilibrium points are located in the (x, y)-plane. The scrolls generated from the generalized nonlinearities can be increased in the x and y state variable directions.
- *3-D grid scroll attractor family:* This is the most complete class of the presented scroll grid attractor families. The equilibrium points are located in 3-D and the system has three nonlinear functions. Due to the generalization of each nonlinearity, the scrolls can be generated in all state variable directions.

In this Section, the main contribution is to show the possibility of generating the equilibrium points on a plane or in 3-D instead of on a line. As a result, it is possible to increase the number of scrolls into all state variable directions. In the literature, a quad screw attractor [125] from a 4-D chaotic oscillator with hysteresis [221] is comparable with a 2 × 2-scroll grid attractor, which is a member of the 2-D grid scroll attractor family. However, it should be noted that the system presented here is simpler than the other one. Moreover, the 2 × 2-scroll grid attractor is just one member of the scroll grid attractor family. It will be shown that this family can be extended by adding a simple nonlinearity. Another related attractor family are n-double scroll hypercubes [257] which occurs in weak unidirectional or diffusive coupling of n-double scroll cells within 1-D CNN. However, this family produces hyperchaotic behavior and the order of the system is much higher than the third order circuit introduced in this section. We will introduce n-double scroll hypercubes in Section 3.4.2. From a circuit design point of view, the new circuit is easily realized by using simple comparators. Moreover, it is possible to systematically increase the complexity of the circuit, by simply using additional core nonlinearities. From an application point of view, this system can produce more complicated signals. Hence it is promising in many applications for chaotic systems.

3.3.1 A new family of n-scroll attractors

In 1992, Genesio and Tesi [91] developed a chaotic system in order to examine the harmonic balance method (see Section 3.1.1.2) for determination of the existence and location of chaotic behavior. Genesio and Tesi [91] successfully applied the harmonic balance method and they have shown that the model exhibits chaotic behavior [91]. In fact, the model was first studied in [53, 11] and [12] with a cubic nonlinearity and a 3-segment piecewise linear nonlinearity, respectively. In [47], it has been shown that this model is qualitatively similar to the dynamics of Chua's oscillator. Recently, Elwakil and Kennedy [78] have reported that this model is extremely simple from a circuit design point of view. In order to have a single bifurcation parameter, they have also introduced a modified version of the model which is described by

$$\begin{cases} \dot{x} &= y \\ \dot{y} &= z \\ \dot{z} &= -ax - ay - az + af(x) \end{cases} \tag{3.56}$$

where

$$f(\zeta) = \begin{cases} 1 & \zeta \geq 0 \\ -1 & \zeta < 0, \end{cases} \tag{3.57}$$

and $x, y, z \in \mathbb{R}$, $\zeta \in \mathbb{R}$. A double-scroll-like attractor has been reported for $a = 0.8$ by Elwakil and Kennedy [78, 77]. A generalization of this original model for generating n-scrolls has been recently given by Yalçın *et al.* [313]. Here, we consider a minor modification of the latter model by taking the nonlinearity

$$f(x) = \sum_{i=1}^{M_x} g_{\frac{(-2i+1)}{2}}(x) + \sum_{i=1}^{N_x} g_{\frac{(2i-1)}{2}}(x) \tag{3.58}$$

where

$$g_\theta(\zeta) = \begin{cases} 1, & \zeta \geq \theta \quad \theta > 0 \\ 0, & \zeta < \theta \quad \theta > 0 \\ 0, & \zeta \geq \theta \quad \theta < 0 \\ -1, & \zeta < \theta \quad \theta < 0. \end{cases} \tag{3.59}$$

With this modification, we obtain a nonlinearity which is parameterized in less unknowns. A computer simulation for the double scroll attractor is shown in Figure 3.19(a) corresponding to $M_x = 0$, $N_x = 1$ and $a = 0.8$. A generalization of the system (3.56) can be systematically obtained by introducing additional breakpoints in the nonlinearity where each breakpoint can be implemented by equation (3.59). Therefore, we call Eq. (3.59) the core function. The equilibrium points can be found from the following set of equations

$$\begin{cases} x = f(x) \\ y = 0 \\ z = 0. \end{cases} \tag{3.60}$$

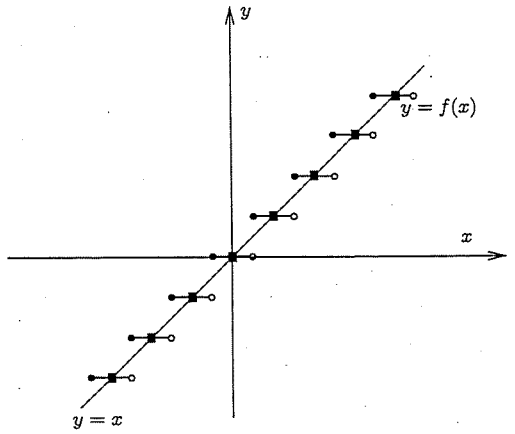


Figure 3.18: 1-D grid scroll attractors: the equilibrium points (squares) are shown at the intersection of x and $f(x)$.

The equilibrium points are located at the intersection of the nonlinear function $f(x)$ and x drawn in Figure 3.18 for the general case. The set of equilibrium points is given by

$$\mathcal{V}_{eq} = \{[i \;\; 0 \;\; 0]^T \,|\, i = -M_x, -M_x + 1, ..., N_x - 1, N_x\} \tag{3.61}$$

which are located on the $x-$axis in state space (Figure 3.26(a)). Due to the location of the equilibrium points, this strange attractor family is called 1-D grid scroll attractors. In the attractors, the scrolls are located around the equilibrium points. Therefore, the number of scrolls equals the number of equilibrium points. The number of scrolls generated from the generalized nonlinearity is equal to $M_x + N_x + 1$. In Figure 3.19, 3-scroll, 5-scroll and 10-scroll attractors are shown by using the generic model (3.56) with Eq. (3.58) for $a = 0.4$ and for $(M_x = 1, N_x = 1)$, $(M_x = 0, N_x = 4)$, $(M_x = 4, N_x = 5)$, respectively. Figure 3.20 shows the bifurcation phenomenon with respect to the parameter value a related to a 5-scroll attractor.

3.3.2 2-D grid scroll attractors

Now we show that it is also possible to increase the number of scrolls in the y state variable direction. We start from the same system (3.56) but with an

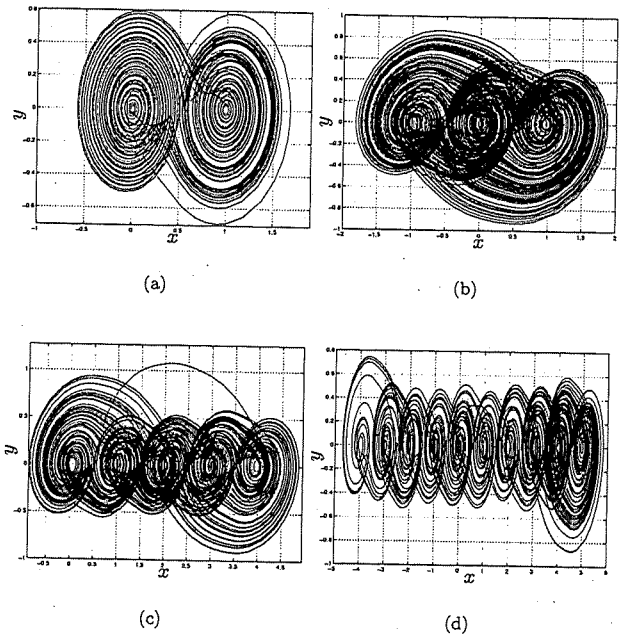


Figure 3.19: (a) $M = 0$, $N = 1$, 2-scroll attractor (double scroll attractor), (b) $M = 1$, $N = 1$, 3-scroll attractor, (c) $M = 0$, $N = 4$, 5-scroll attractor, (d) $M = 4$, $N = 5$, 10-scroll attractor.

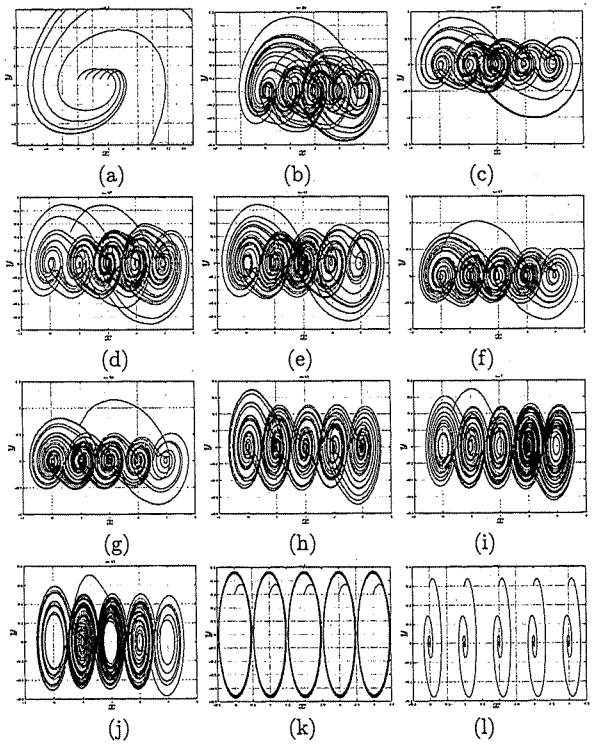


Figure 3.20: Bifurcations related to 5−scroll attractors with respect to parameter value a: (a) $a = 0.1$, for five different initial condition which are close to the equilibrium points; (b) $a = 0.23$, (c) $a = 0.34$, (d) $a = 0.37$, (e) $a = 0.41$, (f) $a = 0.47$, (g) $a = 0.5$, (h) $a = 0.61$, (i) $a = 0.7$, (j) $a = 0.81$, for five different initial conditions which are close to the equilibrium points; (k) $a = 1$, (l) $a = 2$.

additional nonlinearity. Hence the model is described by

$$\begin{cases} \dot{x} &= y - f_1(y) \\ \dot{y} &= z \\ \dot{z} &= -ax - ay - az + af_2(x) \end{cases} \tag{3.62}$$

where

$$f_1(y) = \sum_{i=1}^{M_y} g_{\frac{(-2i+1)}{2}}(y) + \sum_{i=1}^{N_y} g_{\frac{(2i-1)}{2}}(y) \tag{3.63}$$

and additional nonlinearity

$$f_2(x) = \sum_{i=1}^{m-1} \beta g_{p_i}(x) \tag{3.64}$$

where

$$\begin{aligned} p_i &= M_y + 0.5 + (i-1)(M_y + N_y + 1) \\ \beta &= M_y + N_y + 1. \end{aligned} \tag{3.65}$$

The equilibrium points follow from

$$\begin{cases} x + y = f_2(x) \\ y = f_1(y) \\ z = 0. \end{cases} \tag{3.66}$$

The solutions of the second equation of (3.66) has already been solved in the previous section, which was given by

$$u^{eq,y} = \{-M_y, \cdots, -1, 0, 1, \cdots, N_y\}.$$

The points for the x state variable corresponding to each $u_j^{eq,y}$ are determined in a graphical way from the first equation of (3.66). The set of equilibrium points becomes

$$\begin{aligned} \mathcal{V}_{eq} = \{[(i-1)(M_y + N_y + 1) + j \quad -j \quad 0]^T \,|\, i = 1, 2, ..., m; \\ j = -N_y, ..., -1, 0, 1, ..., M_y\}. \end{aligned} \tag{3.67}$$

It should be noted that the locations of the equilibrium points are located in the (x, y)-plane (Figure 3.26(b)) and the system has a number of $m(M_y + N_y + 1)$ equilibrium points. For this reason, we will call this strange attractor family $m \times (M_y + N_y + 1)$-scroll grid attractors. Some resulting 2-D grid scroll attractors and their corresponding nonlinearities are given by:

- 2×2-*grid scroll attractor* ($M_y = 0$, $N_y = 1$, $m = 2$) (Figure 3.21)

$$\begin{aligned} f_1(y) &= g_{0.5}(y) \\ f_2(x) &= 2g_{0.5}(x) \end{aligned}$$

- 2×3-*grid scroll attractor* ($M_y = 0$, $N_y = 2$, $m = 2$) (Figure 3.22)

$$\begin{aligned} f_1(y) &= g_{0.5}(y) + g_{1.5}(y) \\ f_2(x) &= 3g_{0.5}(x) \end{aligned}$$

- 3×3-*grid scroll attractor* ($M_y = 0$, $N_y = 2$, $m = 3$)

$$\begin{aligned} f_1(y) &= g_{0.5}(y) + g_{1.5}(y) \\ f_2(x) &= 3(g_{0.5}(x) + g_{3.5}(x)) \end{aligned}$$

- 4×4-*grid scroll attractor* ($M_y = 0$, $N_y = 3$, $m = 4$) (Figure 3.23(a))

$$\begin{aligned} f_1(y) &= g_{0.5}(y) + g_{1.5}(y) + g_{2.5}(y) \\ f_2(x) &= 4(g_{0.5}(x) + g_{4.5}(x) + g_{8.5}(x)) \end{aligned}$$

- 4×5-*grid scroll attractor* ($M_y = 0$, $N_y = 4$, $m = 4$) (Figure 3.23(b))

$$\begin{aligned} f_1(y) &= g_{0.5}(y) + g_{1.5}(y) + g_{2.5}(y) + g_{3.5}(y) \\ f_2(x) &= 5(g_{0.5}(x) + g_{5.5}(x) + g_{10.5}(x)) \end{aligned}$$

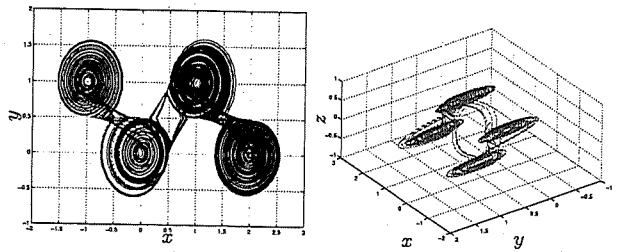
3.3.3 3-D grid scroll attractors

The following additional nonlinearity $f_1(z)$ is introduced now to the system (3.62) which becomes

$$\begin{cases} \dot{x} &= y - f_1(y) \\ \dot{y} &= z - f_1(z) \\ \dot{z} &= -ax - ay - az + af_3(x) \end{cases} \tag{3.68}$$

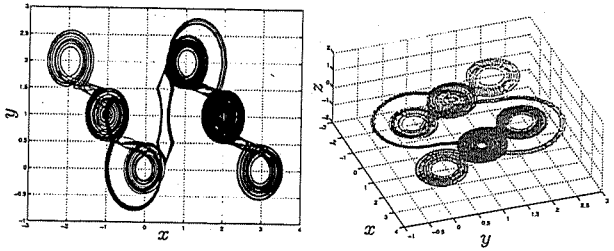
where

$$f_1(z) = \sum_{i=1}^{M_z} g_{\frac{(-2i+1)}{2}}(z) + \sum_{i=1}^{N_z} g_{\frac{(2i-1)}{2}}(z), \tag{3.69}$$



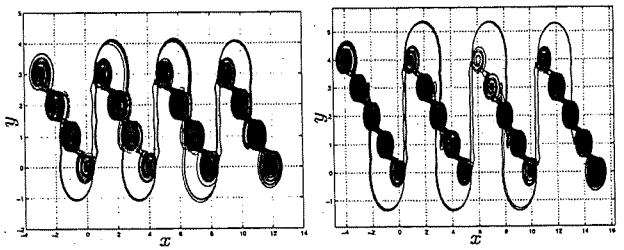
(a) (b)

Figure 3.21: 2×2-grid scroll attractor. Projection onto the (a) (x, y) and (b) view on 3-D state space.



(a) (b)

Figure 3.22: 2×3-grid scroll attractor. Projection onto the (a) (x, y) and (b) view on 3-D state space.



(a) (b)

Figure 3.23: Projection onto the (x, y)-plane of (a) 4×4- and (b) 4×5-grid scroll attractor.

and

$$f_3(x) = \sum_{l=1}^{k-1} \gamma g_{n_l}(x) \tag{3.70}$$

where

$$\begin{aligned} n_l &= \rho + 0.5 + (l-1)(\rho + \varsigma + 1) \\ \gamma &= \rho + \varsigma + 1 \end{aligned} \tag{3.71}$$

with

$$\begin{aligned} \rho &= \left|\min_{i,j}\{u_i^{eq,y} + u_j^{eq,z}\}\right|, \\ \varsigma &= \left|\max_{i,j}\{u_i^{eq,y} + u_j^{eq,z}\}\right| \end{aligned} \tag{3.72}$$

and $u^{eq,y}$ and $u^{eq,z}$ are the vectors for the y and z variable related to the equilibrium points. The equilibrium points follow from

$$\begin{cases} x + y + z = f_3(x) \\ y = f_1(y) \\ z = f_1(z) \end{cases} \tag{3.73}$$

where the points for the y, z variable are given by are

$$\begin{aligned} u^{eq,y} &= \{-M_y, \cdots, -1, 0, 1, \cdots, N_y\} \\ u^{eq,z} &= \{-M_z, \cdots, -1, 0, 1, \cdots, N_z\}. \end{aligned} \tag{3.74}$$

With these nonlinearities the system produces $k \times (M_y + N_y + 1) \times (M_z + N_z + 1))$-scroll grid attractors. All the scrolls are located around the equilibrium points which are given

$$\begin{aligned} \mathcal{V}_{eq} = \{[(l-1)(\varsigma + 1 + \rho) - u_i^{eq,y} - u_j^{eq,z} \;\; u_i^{eq,y} \;\; u_j^{eq,z}]^T \,|i = 1, 2, ... \\ M_y + N_y + 1; j = 1, 2, ...M_z + N_z + 1; \;\; l = 1, 2, ..., k\}. \end{aligned} \tag{3.75}$$

The location of the equilibrium points in 3-D is shown in Figure 3.26(c). Here, some examples of 3-D grid scroll attractors are given:

- $2 \times 2 \times 2$-*grid scroll attractor* ($M_y = 0$, $N_y = 1$, $M_z = 0$, $N_z = 1$, $k = 2$) (Figure 3.24)

$$f_3(x) = 3g_{0.5}(x)$$
$$f_1(y) = g_{0.5}(y)$$
$$f_1(z) = g_{0.5}(z)$$

- $4 \times 2 \times 2$-*grid scroll attractor* ($M_y = 0$, $N_y = 1$, $M_z = 0$, $N_z = 1$, $k = 4$) (Fig. 3.25)

$$f_3(x) = 3(g_{0.5}(x) + g_{3.5}(x) + g_{6.5}(x))$$
$$f_1(y) = g_{0.5}(y)$$
$$f_1(z) = g_{0.5}(z)$$

- $4 \times 3 \times 2$-*grid scroll attractor* ($M_y = 1$,$N_y = 1$,$M_z = 0$, $N_z = 1$, $k = 4$) (Figure 3.25)

$$f_3(x) = 4(g_{1.5}(x) + g_{5.5}(x) + g_{9.5}(x))$$
$$f_1(y) = g_{-0.5}(y) + g_{0.5}(y)$$
$$f_1(z) = g_{0.5}(z)$$

3.3.4 Circuit realizations

In this section, the realizations of some of the 1-D, 2-D, and 3-D grid scroll attractors discussed above are given. For this purpose, a circuit using conventional voltage opamps could be employed. However, a number of research results, which illustrate the advantages of current feedback opamps (CFOA) over conventional voltage opamps, have been presented in the literature, e.g. [268, 80, 267]. From these works, it is known that CFOA is almost free from slew-rate limitations as opposed to voltage opamp. It is capable of operating at much higher frequencies and offers design flexibility which allows the derivation of relatively simpler circuits. Considering these facts, researchers have attempted to use CFOAs in the implementation of chaotic oscillators in

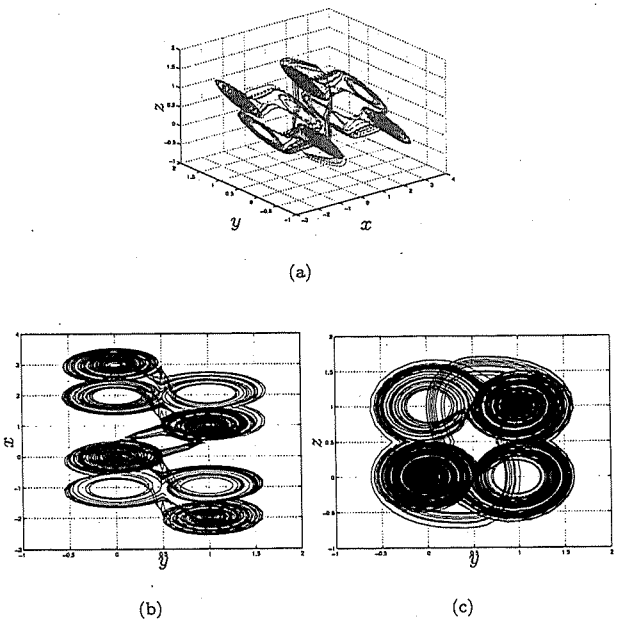


Figure 3.24: $2 \times 2 \times 2$-grid scroll attractor. (a) View on 3-D state space. Projection onto the (b) (y, x) and (c) (y, z)-plane.

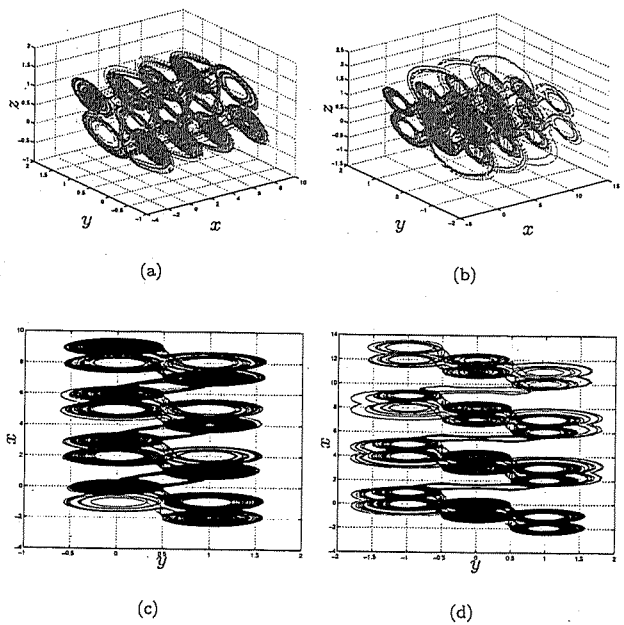


Figure 3.25: $4 \times 2 \times 2$-grid scroll attractor and $4 \times 3 \times 2$-grid scroll attractor: (a)-(b) view on 3-D state space, projection onto the (c)-(d) (y, x), respectively.

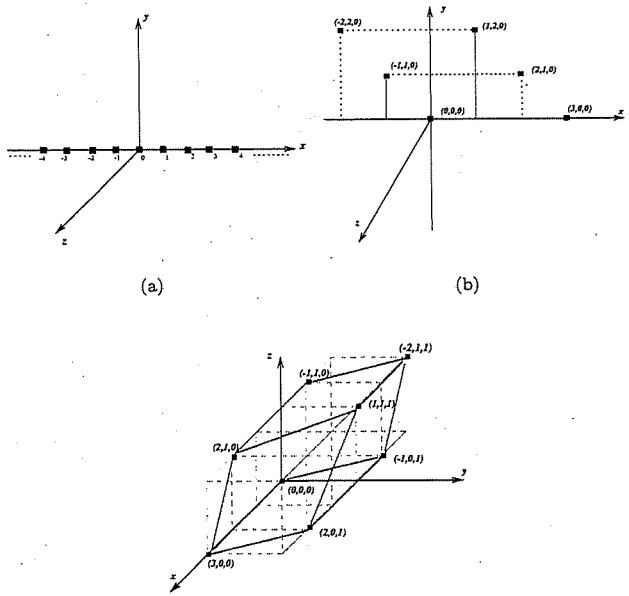


Figure 3.26: The equilibrium points are located on (a) $x-$axis for 1-D grid scroll attractor, (b) (x, y)-plane for 2-D grid scroll attractor (equilibrium points are shown for 2×3-grid attractor), (c) in a 3-D body for 3-D grid scroll attractors (equilibrium points are shown for a $2 \times 2 \times 2$-scroll grid attractor).

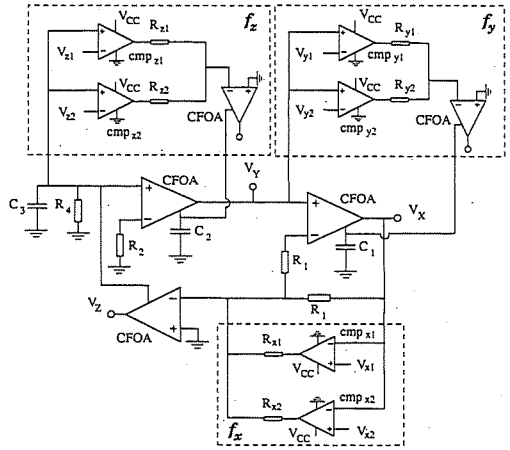


Figure 3.27: Possible circuit diagram realizing the proposed system.

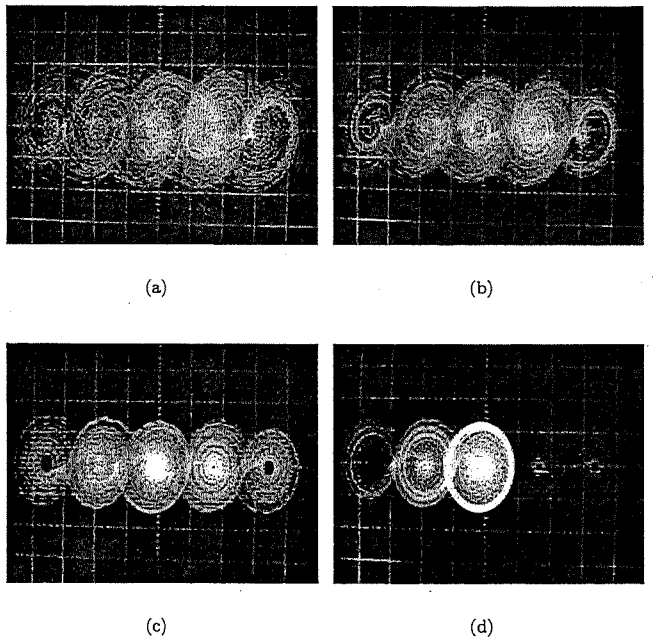
order to have an improved high-frequency performance, e.g. [230, 75, 76]. In the same way, we have also used a possible CFOA-based circuit in order to realize and observe some of the scroll grid attractors described in the previous sections. The circuit that we have used for this purpose is given in Figure 3.27. The CFOAs are implemented using AD844[2] from Analog Devices and the required nonlinearities in Eqs. (3.63), (3.69) and (3.70) are realized using the sub-circuits drawn within the dashed lines. The comparators (*cmps.*) involved in these sub-circuits are of LM311 type comparators. The sub-circuits f_x, f_y and f_z within the dashed lines are used to increase the number of scrolls in the x, y and z state variable directions, respectively. Since each of these sub-circuits includes two LM311 comparators, a $3 \times 3 \times 3$-grid scroll attractor or 3-D grid scroll attractor can be observed using this circuit. Obviously, by appropriately inserting additional comparators in the corresponding sub-circuit, the number of scrolls can be systematically increased in all directions. Also, by appropriately removing these sub-circuits, new circuits allowing the observation of any 1-D and 2-D grid scroll attractor can readily be obtained. For $C_1 = C_2 = C_3 = C$, $R_1 = R$, $R_2 = R_4 = R/a$, $V_x = ax$, $V_y = ay$, $V_z = z$ and using the normalized quantity $t_n = t/RC$, it can be verified that the circuit realizes the system in Eq. (3.56). Also, as explained above, the nonlinearities in Eqs. (3.63), (3.69) and (3.70) are realized using the sub-circuits f_y, f_z and f_x,

[2]Analog Devices [1990] "Linear products data book," *Norwood, Massachusetts, USA.*

respectively, where the parameters θ of the core functions, $g_\theta(\cdot)$ in Eq. (3.82) are adjusted through the controlling voltages at the inverting/non-inverting inputs of the comparators. It should be noted that all three states are available at the buffered output terminals of the CFOAs, a property which is expected to simplify the realizations of various chaotic communication systems based on the proposed circuit. Also, the fact that all the capacitors are grounded simplifies the IC integration of the circuit. In all the experiments given in this section, CFOAs and LM311 type comparators are supplied under $\pm 15V$ DC and the passive component values are taken as $C_1 = C_2 = C_3 = 1nF$, $R_1 = 5.1k\Omega$. For all experiments, different values have been assigned to the other passive components. Also, the controlling voltages at the inverting and non-inverting inputs of the comparators are taken as adjustable.

First, we have implemented a 5-scroll attractor or 1-D grid scroll attractor from the circuit in Figure 3.27. In order to obtain a 5-scroll attractor in the x state variable direction, we have removed the sub-circuits f_y and f_z and added two more comparators to the sub-circuit f_x. The passive component values are taken as $R_2 = R_4 = 8k\Omega$, $R_{x1} = R_{x2} = R_{x3} = R_{x4} = 70k\Omega$. The observed (V_x, V_y) trajectory is given in Figure 3.28(c). These passive component values correspond to $a = 0.64$. Also, by changing the values of the resistors R_2 and R_4, the circuit is tested for different values of a, and the corresponding results are given in Figure 3.28(a)-(d). These results also verify the dynamic behavior of the circuit with respect to parameter value a, which was shown on the simulations of Fig. 3.20.

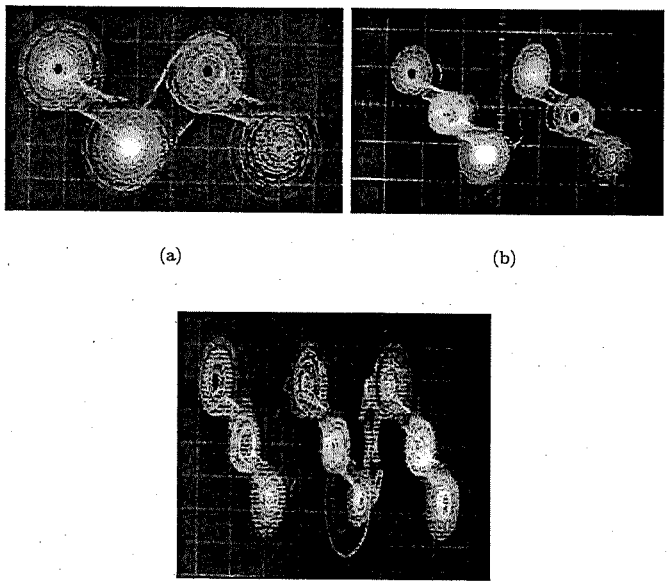
Second, we have implemented 2×2-, 3×2- and 3×3-grid scroll attractors from 2-D grid scroll attractor family. In order to have a 2×2-grid scroll attractor, we have removed the sub-circuit f_z, the comparators cmp_{x2}, cmp_{y2} and the resistors R_{x2}, R_{y2}, from the circuit in Fig. 3.27. The passive components values are taken as $R_2 = R_4 = 9.7k\Omega$, $R_{x1} = 37k\Omega$, $R_{y1} = 77k\Omega$. The controlling voltages, i.e. V_{x1}, V_{y1} are taken identical. The observed phase space corresponding to the (V_x, V_y) trajectory is given in Figure 3.29(a). A 3×2—grid scroll attractor is realized by adding a comparator, cmp_{y2} and a resistor R_{y2} to the circuit above which was used to realize 2×2-grid scroll attractor. The passive components values are taken as $R_2 = R_4 = 11k\Omega$, $R_{x1} = 27k\Omega$, $R_{y1} = 88k\Omega$, $R_{y2} = 75k\Omega$. The controlling voltages V_{x1}, V_{y1} are taken identical. The observed (V_x, V_y) trajectory shown in Figure 3.29(b) verifies the theory. As a final example of the 2-D grid attractor family, we have realized a 3×3— grid attractor by increasing the number of comparators in the sub-circuit f_x by one. In this case, the passive component values are chosen as $R_2 = R_4 = 12k\Omega$, $R_{x1} = 28k\Omega$, $R_{x2} = 30k\Omega$, $R_{y1} = 90k\Omega$, $R_{y2} = 80k\Omega$. Again, the measured (V_x, V_y) trajectory is given in Figure 3.29(c).



(a) (b)

(c) (d)

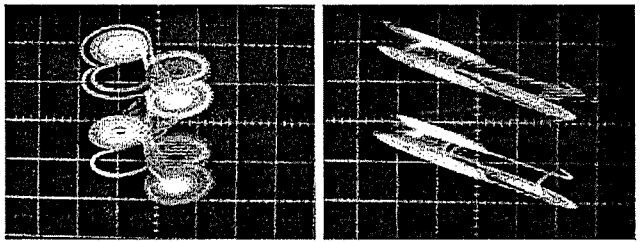
Figure 3.28: 5-scroll attractor from a 1-D grid scroll attractor. Experimental results shown are for (a) $a = 0.26$ ($R_4 = 20k\Omega$), (b) $a = 0.34$ ($R_4 = 15k\Omega$), (c) $a = 0.64$ ($R_4 = 8k\Omega$), (d) $a = 1$ ($R_4 = 5.1k\Omega$) , (V_x,V_y) trajectory $x = 0.5V/div$, $y = 0.5V/div$.



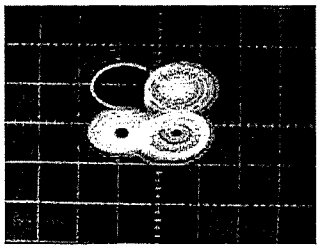
(a) (b)

(c)

Figure 3.29: (a) 2 × 2-grid, (b) 2 × 3-grid, (c) 3 × 3-grid scroll attractor. Experimental results shown are (V_x,V_y) trajectory. $x = 0.5V/div$, $y = 0.5V/div$, $x = 1V/div$, $y = 0.5V/div$ and $x = 1V/div$, $y = 0.5V/div$, respectively.



(a) (b)



(c)

Figure 3.30: $2 \times 2 \times 2$-grid scroll attractor. Experimental results shown are (a) (V_y,V_x) trajectory. $x = 1V/div$, $y = 2V/div$, (b) (V_z,V_x) trajectory. $x = 1V/div$, $y = 2V/div$, (c) (V_y,V_z) trajectory. $x = 1V/div$, $y = 2V/div$

Finally, a $2 \times 2 \times 2$-grid attractor from the 3-D grid attractor family is realized using the circuit in Fig. 3.27. In order to have a $2 \times 2 \times 2-$grid scroll attractor with $M_y = 0$, $N_y = 1$, $M_z = 0$, $N_z = 1$ and $k = 2$, we have removed the comparators cmp_{x2}, cmp_{y2} and cmp_{z2} in the sub-circuits within the dashed lines and the passive component values are taken as $R_2 = R_4 = 8.3k\Omega$, $R_{x1} = 19k\Omega$, $R_{y1} = 47k\Omega$, $R_{z1} = 50k\Omega$. The controlling voltages at the comparator inputs are taken identically. The experimental results corresponding to the (V_y, V_x), (V_z, V_x) and (V_y, V_z) trajectories are given in Figure 3.30.

3.3.5 Alternative realizations

Two new circuit realizations of a double scroll attractor from the model (3.56) have been recently proposed based on MOS-only [202] and log domain PNP-only transistors [193]. The latter are very suitable for low-voltage and high frequency operation and can also be implemented in integrated circuits. An alternative realization of 1-D grid scroll attractors from the model (3.56) with smooth hyperbolic tangent functions ($\tanh(\cdot)$) instead of the nonlinearity (3.58) has been presented by Özoğuz *et al.* [192]. Furthermore, its implementation only requires properly biased bipolar differential-pair cells. Therefore, the structure of the proposed circuit is much simpler than for existing n-scroll attractors circuits. The proposed nonlinearity is described by

$$f(x) = x - \sum_{i=-N}^{M} (-1)^{i-1} \tanh \gamma(x - o_i) \tag{3.76}$$

where $o_i = 2i$, M and N are odd integers. Özoğuz *et al.* [192] have observed a 5-scroll attractor in the system (3.56) using the nonlinearity (3.76) for $a = 0.25$, $M = 5$, $N = 3$, $\gamma = 2$. In [222], the hyperbolic tangent function ($\tanh(\cdot)$) is replaced with a switching $\text{sgn}(\cdot)$ function in Eq. (3.76). Using the same values, a 5-scroll attractor observed in this case also.

Recently, Yang and Li [317] and Han *et al.* [104] have also proposed a model to generate multi-scroll attractors which are motivated by the families of scroll grid attractors. Han *et al.* [104] have shown that 1-D grid and 2-D grid scroll attractors can be generated in a linear second order continuous system via switching of hysteresis series.

3.4 Multi-scroll hyperchaotic attractors

Hyperchaos is defined as a chaotic attractor with more than one positive Lyapunov exponent [218]. The first physical system which exhibits hyperchaotic

behavior was introduced in [175]. In [175], the following four dimensional circuit (Matsumoto-Chua-Kobayashi (MCK) circuit) has been proposed,

$$\begin{cases} \dot{x} &= \alpha_1 g(y-x) - \alpha_2 z \\ \dot{y} &= \beta[-g(y-x) - w] \\ \dot{z} &= x + z \\ \dot{w} &= \gamma y \end{cases} \tag{3.77}$$

with piecewise linear characteristic

$$g(y-x) = m_0(y-x) + \frac{1}{2}(m_1 - m_0)(|y-x+1| - |y-x-1|). \tag{3.78}$$

For $\alpha_1 = \alpha_2 = 2$, $\beta = 20$, $m_0 = 3$ and $m_1 = -0.2$ one obtains hyperchaos as reported in [175].

3.4.1 Hyperchaotic n-scroll attractors

In this section, we show that how the MCK circuit can be generalized by taking additional breakpoints in PWL characteristic, in a similar way as the Chua's circuit has been modified to generate n-scroll attractors. The resulting generalized MCK circuit shows more complicated hyperchaotic behavior than the original one. The PWL characteristic by introducing additional breakpoints is described by

$$g(y-x) = m_0(y-x) + \frac{1}{2}\sum_{i=1}^{2q-1}(m_i - m_{i-1})(|y-x+c_i| - |y-x-c_i|) \tag{3.79}$$

where $m = [m_0\ m_1 \cdots m_{2q-1}]$, $c = [c_1\ c_2 \cdots c_{2q-1}]$ and n_h denotes a natural number [303, 252]. As a function of the number of breakpoints in the PWL characteristic one obtains:

- $n_h = 1$: This case is the same as (3.78)
- $n_h = 2$: Hyperchaotic 4-scroll attractor, shown in Figure 3.31 with parameters

  $\alpha_1 = 2.4$, $\alpha_2 = 2$, $\beta = 20$, $\gamma = 1.5$
  $m = [3\ \ -0.1\ \ 4.3\ \ 0.2]$, $c = [1\ \ 2.2\ \ 4.3]$.

- $n_h = 3$: Hyperchaotic 6-scroll attractor, shown in Figure 3.32 with parameters

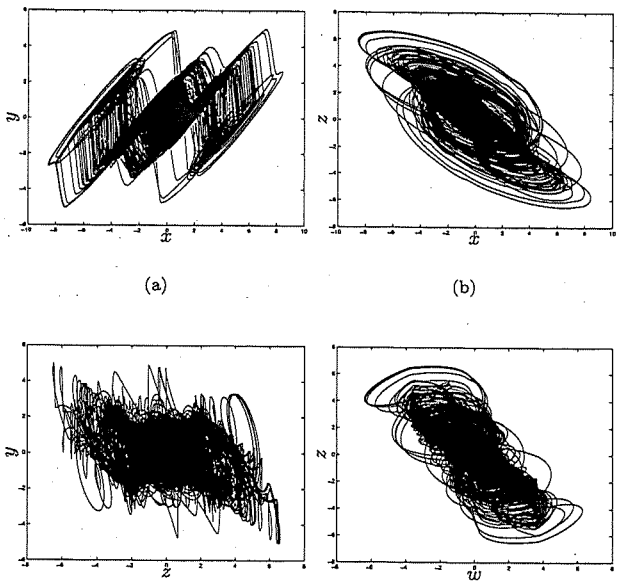


Figure 3.31: Hyperchaotic 4-scroll attractor observed by computer simulation for $x(0) = 3$, $y(0) = -0.1$, $z(0) = 4.3$, $w(0) = -0.2$: Projection onto the (a) (x, y), (b) (x, z), (c) (z, y), (d) (w, z)-plane.

$\alpha_1 = 2.4$, $\alpha_2 = 2$, $\beta = 20$, $\gamma = 1.5$
$m = [3\ \ -0.1\ \ 4.3\ \ -0.04\ \ 6.1\ \ -0.15]$, $c = [1\ \ 2.2\ \ 4.3\ \ 7.4\ \ 10]$.

3.4.2 n-Scroll hypercube attractors

Here we will show that how the CNN framework can be exploited in order to generate hyperchaos. Kapitaniak *et al.* [124, 123] investigated the hyperchaotic attractors in an open and closed chain of coupled identical Chua's circuits that behave individually as double scrolls. This chain of coupled identical Chua's circuit describes a one-dimensional CNN consisting of Chua's circuit cells [257].

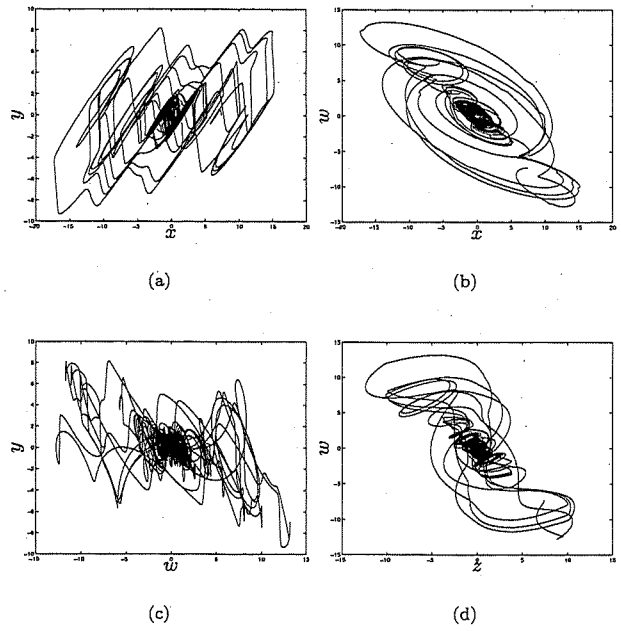


Figure 3.32: Hyperchaotic 6-scroll attractor observed by computer simulation for $x(0) = 3$, $y(0) = -0.1$, $z(0) = 4.3$, $w(0) = -0.2$: Projection onto the (a) (x, y), (b) (x, z), (c) (z, y), (d) (w, z)-plane.

Taking a 1-D CNN consisting of n-double scroll cells with weak unidirectional or diffusive coupling between the cells, an n-double scroll hypercube was obtained in a common state space of the cells [257]. Here, we consider a 1-D CNN consisting of identical generalized Chua's circuits (3.31)-(3.32), with unidirectional coupling between the cells

$$\begin{cases} \dot{x_j} &= \alpha\,[y_j - h(x_j)] \\ \dot{y_j} &= x_j - y_j + z_j + K_{j-1}(y_j - y_{j-1}) \\ \dot{z_j} &= -\beta y_j, \quad j = 1, 2, ...L, \end{cases} \tag{3.80}$$

and with diffusive coupling between the cells

$$\begin{cases} \dot{x_j} &= \alpha\,[y_j - h(x_j)] + D_x(x_{j-1} - 2x_j + x_{j+1}) \\ \dot{y_j} &= x_j - y_j + z_j \\ \dot{z_j} &= -\beta y_j, \quad j = 1, 2, ...L, \end{cases} \tag{3.81}$$

where j denotes the cell number and L is the number of cells. We impose the conditions $y_0 = y_L$ for Eq. (3.80), $x_0 = x_L$ and $x_{L+1} = x_1$ for Eq. (3.81). For coupling constants, $K_0 = 0$, $K_i = K$ $(i = 1, ..., L-1)$ and a positive diffusion coefficient D_x are chosen.

1-D CNNs with unidirectional coupling between the cells has been studied by Kapitaniak and Chua [123] for the case $q = 1$, leading to the double-double scroll attractor. In a similar way, by taking n-scroll attractors as cells one obtains so-called n-scroll squares $(L = 2)$, n-scroll cubes $(L = 3)$ and n-scroll hypercube attractors $(L \geq 4)$ in the common state subspace of the cells [257], i.e. in (x_1, x_2) $(L = 2)$, (x_1, x_2, x_3) $(L = 3)$, (x_1, x_2, x_3, x_4) $(L = 4)$ etc. Figure 3.33 shows a 2-scroll and 5-scroll hypercube attractors $(L = 4)$ in the common state subspace of the cells. Weak coupling has been taken with $K = 0.06$. The initial states are chosen close to the origin. Similar results are obtained for diffusive coupling between the chaotic cells.

3.5 Scroll maps from n-scroll attractors

Recently, chaotic piecewise-linear one-dimensional maps have received considerable attention because of their potential application in communications [220, 219], cryptography [237, 96] and also neural networks [186]. Furthermore, the theory of one dimensional maps [158, 272] is well developed. It would be useful if an appropriate 1-D map can be constructed from the systems under the study. An elegant way for this purpose has been given by Poincaré. Wu and Rul'kov [288] demonstrate that 1-D maps can be used to analyze the dynamic and bifurcation phenomena of the Chua's circuit using the Poincaré section.

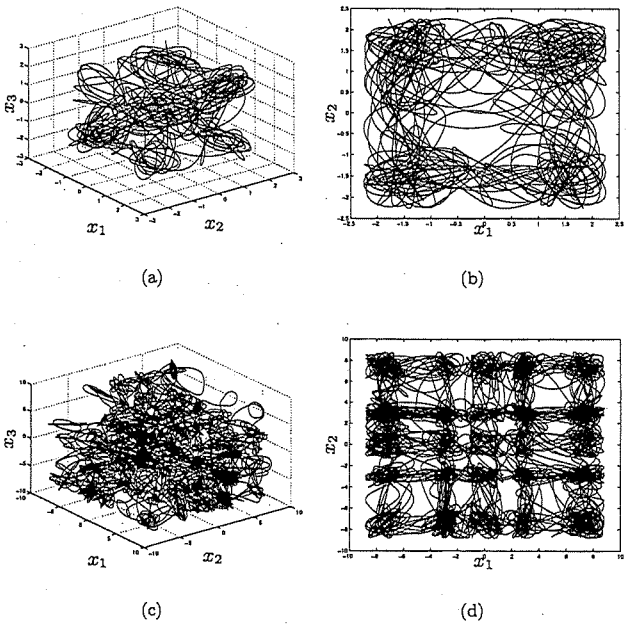


Figure 3.33: n-scroll hypercube attractors: (a)-(b) 2-scroll hypercube attractor and (b)-(c) 5-scroll hypercube attractor from a 1-D CNN with 4 cells of 2-scroll and 5-scroll attractor and unidirectional coupling between cells ($K = 0.06$), respectively. Two views are shown for each attractor: (a) and (c) (x_1, x_2, x_3); (b) and (d) (x_1, x_2).

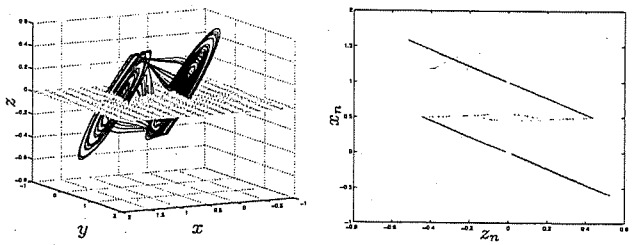
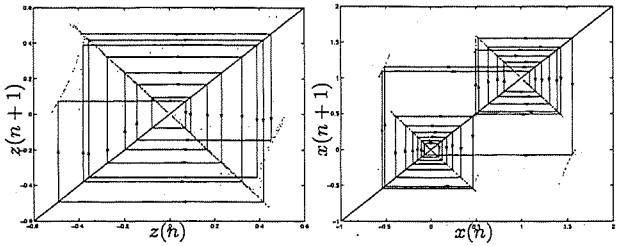


Figure 3.34: (a) A Poincaré section for the double scroll attractor. (b) Poincaré map from the intersection (Figure 3.34(a)).

Here, we present the scroll maps which are imitated behavior of the two-sided Poincaré map of n-scroll attractors.

3.5.1 1-Scroll and 2-scroll maps

A Poincaré section is an $n-1$ dimensional hyperplane which cuts the state space trajectory of a dynamical system in $\mathbb{R}^n$ [195]. A map from $\mathbb{R}^{n-1}$ to $\mathbb{R}^{n-1}$ can be obtained from this dynamical system. Given a point on the Poincaré section, the Poincaré map maps this point to the next point on the trajectory through this point which intersect the Poincaré section in the positive and negative directions [195]. Figure 3.34(a) shows a 2-double scroll attractor from the system (3.56) and its generalized nonlinearity (3.58) with $a = 0.8$ and $M_x = 0$, $N_x = 1$, intersecting the plane U transversally. Here, we define a Poincaré section as $U = \{[x\ y\ z]^T : [0\ 1\ 0][x\ y\ z]^T = 0\}$, hence a two-sided Poincaré map can be obtained from $\mathbb{R}^2$ into $\mathbb{R}^2$. The corresponding Poincaré map of this section is shown in Figure 3.34(b) which shows a 2-D map. 1-D maps can be constructed from this 2-D map by means of a projection. In the Figures 3.35(a) and (b) two constructed map from the two-sided Poincaré map are shown by plotting the orbits (z_n, z_{n+1}) and (x_n, x_{n+1}) respectively. Graphical iteration of the maps has been also shown on the Figures 3.35(a) and (b) which lead us the names of to 1− and 2−scroll map, respectively. Similar observations can be made for n-scroll attractors by using the same Poincaré section. Let us now define a similar 1-D map by $x_{n+1} = f_P(x_n)$ where $f_P : [0,1] \rightarrow [0,1]$. First, we



(a) (b)

Figure 3.35: (a) 1-scroll map and (b) 2-scroll map from the projection via the the Poincaré map (Figure 3.34(b)). The different iteration steps are illustrated.

propose a 1-*scroll map* described by

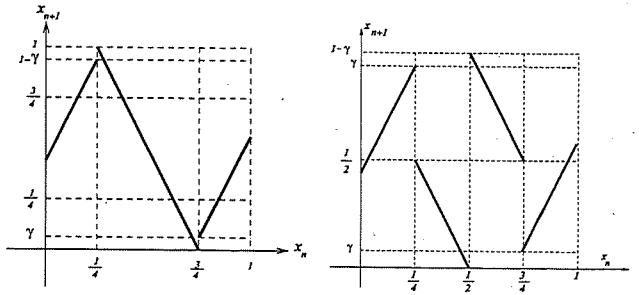
$$x_{n+1} = \begin{cases} 2(x_n - \frac{1}{8}) + \frac{3}{4} + \gamma, & 0 \leq x_n < \frac{1}{4} \\ 2(\frac{1}{2} - x_n) + \frac{1}{2}, & \frac{1}{4} \leq x_n \leq \frac{3}{4} \\ 2(x_n - \frac{7}{8}) + \frac{1}{4} + \gamma, & \frac{3}{4} < x_n \leq 1 \end{cases} \tag{3.82}$$

with $\gamma \in (0, \frac{1}{2})$ and shown in Figure 3.36. The quantification of the system is done by estimating the Lyapunov exponent. Theoretically, the Lyapunov exponent of one-dimensional maps f_P is defined as

$$\lambda := \lim_{N \to \infty} \frac{1}{N} \sum_{n=0}^{N-1} \ln \left| \frac{df_P}{dx}(f_P^{(n)}(x_0)) \right| \tag{3.83}$$

for $x_0 \in [0, 1]$ [272]. One should note that $|\frac{d}{dx}(f_P(x_i))|$ $\forall i$ is equal to 2 in each linear region. Hence, the Lyapunov exponent will be equal to $\ln 2$. We decided to calculate the Lyapunov exponent from the time series [121], whose first 200 points are given for testing this result. TISEAN 2.1 [264], which is a free software project for the analysis of time series with methods based on the theory of nonlinear deterministic dynamical systems, has been used to calculate the largest Lyapunov exponent. The result obtained from this time series indicates that $\lambda = \ln 2$.

Furthermore a 2-scroll map for the same Poincaré section (Figure 3.34) is given



(a) (b)

Figure 3.36: (a) 1-scroll map. (b) 2-scroll map.

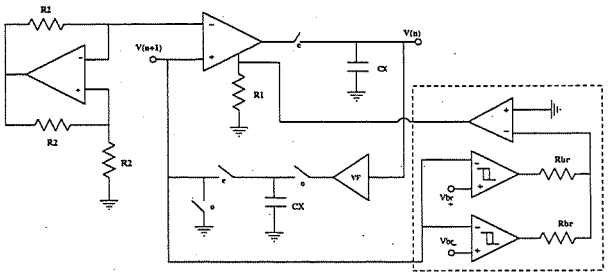
by

$$x_{n+1} = \begin{cases} 2(x_n - \frac{1}{8}) + \frac{3}{4} + \gamma, & 0 \leq x_n < \frac{1}{4} \\ 2(\frac{3}{8} - x_n) + \frac{1}{4}, & \frac{1}{4} \leq x_n < \frac{1}{2} \\ 2(\frac{5}{8} - x_n) + \frac{3}{4}, & \frac{1}{2} \leq x_n < \frac{3}{4} \\ 2(x_n - \frac{7}{8}) + \frac{1}{4} + \gamma, & \frac{3}{4} \leq x_n \leq 1 \end{cases} \tag{3.84}$$

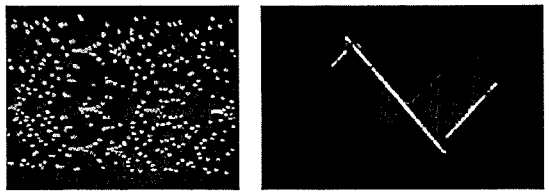
where $\gamma \in (0, \frac{1}{2})$ and shown in Figure 3.36. Again, the Lyapunov exponent of the 2-scroll map for $\gamma = 0.1$ is estimated from the time series based on the first 200 points. The result obtained from this time series indicates that $\lambda = \ln 2$.

3.5.2 Circuit realization of a 1-scroll map

In order to verify the proposed maps, the 1-scroll map of Eq.(3.82) is implemented using discrete components. The circuit for this purpose is shown in Figure 3.37(a). The opamps are all CFOAs. The voltage follower (VF) is also realized using another CFOA. The voltage V_n corresponds to the value of the state x at the n-th discrete time instance. The subcircuit composed of the resistors $R1$ and $R2$s and the associated opamps is used to implement the required nonlinearity. This circuit follows the approach of the robust Chua diode realisation using voltage negative impedance converters [129, 78]. The subcircuit involving the VF and the associated switches implements the delay, whereas the subcircuit within the dashed lines realizes the discontinuity at the breakpoints of the map in Figure 3.36. The voltages Vbr_+ and Vbr_- are taken as adjustable and they correspond with the breakpoints for the map. The slopes of the piecewise linear function, which determine the dynamic behavior

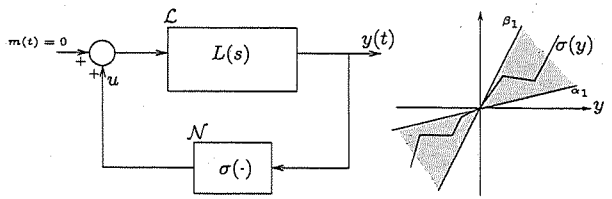


(a)



(b) (c)

Figure 3.37: (a) A circuit realization for the 1-scroll map; (b) Observed time waveform of the signal V_n. $(1V/div, 1V/div)$; (c) (V_{n+1}, V_n) for the 1-scroll map $(0.6V/div, 1V/div)$.



(a) (b)

Figure 3.38: Lur'e system consisting of a linear dynamical system interconnected by feedback to a nonlinearity $\sigma(y)$ (for $n_h = 1$) in sector $[\alpha_1, \beta_1]$.

of the system can be adjusted by the ratio $R1/R2$. The values of $R1$ and $R2$s are $15k\Omega$ and $8.2k\Omega$, respectively, which correspond to a slope equal to 1.83. The capacitors employed in the delay subcircuit are chosen as $50pF$ and all active elements are supplied under $\pm 5V$ DC. The CFOAs are implemented using AD844 IC from Analog Devices, the switches are from 4016 IC chip, the comparators are LM311 IC and the clock frequency is taken as $50kHz$. The measured waveform for the signal V_n is shown in Figure 3.37(b). Note from this figure that V_n is a return-to-zero type signal. Also, the observed phase-space corresponding to (V_{n+1}, V_n) voltages is shown in Figure 3.37(c). These observations verify the feasibility of the proposed system.

3.6 Lur'e representations

The system

$$\begin{cases} \dot{x} &= Ax + Bu \\ y &= Cx \\ u &= \sigma(y), \end{cases} \tag{3.85}$$

with state vector $x \in \mathbb{R}^n$, $u, y \in \mathbb{R}^{n_h}$ and matrices $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times n_h}$, $C \in \mathbb{R}^{n_h \times n}$, describes a *Lur'e system*, a classical structure in control theory. It corresponds to the feedback block diagram of Figure 3.38. This represents a linear dynamical system $(\mathcal{L})$, interconnected by feedback to a static nonlinearity $(\mathcal{N})$ $\sigma(\cdot) : \mathbb{R}^{n_h} \rightarrow \mathbb{R}^{n_h}$ that is required to satisfy a *sector condition*. The following definition of a sector nonlinearity covers many nonlinearities of interest.

Definition 3.1 (Sector Condition) [134] *A memoryless nonlinearity* $\sigma(\cdot) : \mathbb{R}^{n_h} \rightarrow \mathbb{R}^{n_h}$ *is said to satisfy a sector condition* $[K_{min}, K_{max}]$ *if*

$$[\sigma(y) - K_{min}y]^T[\sigma(y) - K_{max}y] \leq 0, \ \forall\, t \geq 0, \ \forall y \in \Gamma \subset \mathbb{R}^{n_h} \tag{3.86}$$

for given real matrices $K_{min} = \text{diag}(\alpha_1, \alpha_2, \cdots, \alpha_{n_h})$ *and* $K_{max} = \text{diag}(\beta_1, \beta_2, \cdots, \beta_{n_h})$, *where* $K = K_{max} - K_{min}$ *is a symmetric positive definite matrix and the interior of* Γ *is connected and contains the origin. If* $\Gamma = \mathbb{R}^{n_h}$, *then* $\sigma(\cdot)$, *satisfies the sector condition globally, in which case it is said that* $\sigma(\cdot)$ *belongs to a sector* $[K_{min}, K_{max}]$. *If (3.86) holds with strict inequality, then* $\sigma(\cdot)$ *is said to belong to a sector* (K_{min}, K_{max}).

In [244], Lur'e systems were interpreted as a class of recurrent neural networks of the form

$$\dot{x} = Ax + B\sigma(Cx), \tag{3.87}$$

with interconnection matrices A, B, C having one hidden layer with a number of n_h hidden neurons and zero bias terms. The static nonlinearity $\sigma(Cx)$ is a diagonal nonlinearity with $\sigma_i(Cx)$ where $i \in \{1, 2, \ldots n_h\}$ belonging to a sector condition.

A Lur'e representation for the Chua's circuit is

$$A = \begin{bmatrix} -\alpha(m_1+1) & \alpha & 0 \\ 1 & -1 & 1 \\ 0 & -b & 0 \end{bmatrix}, B = \begin{bmatrix} -\alpha\,\Delta m \\ 0 \\ 0 \end{bmatrix}, C^T = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{3.88}$$

where $\Delta m = m_0 - m_1$. If one takes the nonlinearity $\sigma(x_1) = \frac{1}{2}(|x_1+1| - |x_1-1|)$ (linear characteristic with saturation), this belongs to sector $[0, 1]$.

The generalized Chua's circuit (3.31) with multiple breakpoint characteristic (3.32) can be represented in Lur'e form with sector [0,1] nonlinearities by considering additional hidden units ($n_h > 1$)

$$A = \begin{bmatrix} a & \alpha & 0 \\ 1 & -1 & 1 \\ 0 & -\beta & 0 \end{bmatrix}, B^T = \begin{bmatrix} b_1 & 0 & 0 \\ b_2 & 0 & 0 \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ b_{2q-1} & 0 & 0 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ 1 & 0 & 0 \end{bmatrix} \tag{3.89}$$

with $a = -\alpha m_{2q-1}$, $b_i = \alpha(m_{i-1} - m_i)$, and $i = 1, 2, \cdots, n_h = 2q - 1$. The vector valued nonlinearity $\sigma = [\sigma_1 \ \sigma_2 \ \cdots \ \sigma_{2q-1}]^T$ with $\sigma_i(x; c_i) = \frac{1}{2}(|x +$

$c_i| - |x - c_i|)$ where $i = 1, 2, \cdots, n_h = 2q - 1$ belong to sector $[0, 1]$. A Lur'e representation of the chaotic cellular neural network (3.29) with its generalized nonlinearity (3.43) is also given by the same nonlinearity and system matrices (3.89) where $a = -\alpha(1 + \delta + \overline{m}_{2q-1})$ and $b_i = \alpha(\overline{m}_{i-1} - \overline{m}_i)$.

The scroll grid attractor families can be also represented as Lur'e systems with,

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a & -a & -a \end{bmatrix}, B = \begin{bmatrix} b_y & 0 & 0 \\ 0 & b_z & 0 \\ 0 & 0 & a \end{bmatrix}, C = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}. \tag{3.90}$$

One obtains the following for the different cases:

- *1-D scroll grid attractors (n−scroll attractor):*

$$\begin{cases} b_y = 0, \, b_z = 0 \\ \sigma(\cdot) = \begin{bmatrix} 0 \\ 0 \\ f_1(\cdot) \end{bmatrix} \end{cases}$$

where $f_1(\cdot)$ is given by Eq. (3.63), which belongs to sector [0, 2].

- *2-D scroll grid attractors:*

$$\begin{cases} b_y = -1, \, b_z = 0 \\ \sigma(\cdot) = \begin{bmatrix} f_1(\cdot) \\ 0 \\ f_2(\cdot) \end{bmatrix} \end{cases}$$

where $f_2(\cdot)$ is given by Eq. (3.64), which belongs to sector [0, $\frac{M_y+N_y+1}{M_y+0.5}$].

- *3-D grid scroll attractors:*

$$\begin{cases} b_y = -1, \, b_z = -1 \\ \sigma(\cdot) = \begin{bmatrix} f_1(\cdot) \\ f_1(\cdot) \\ f_3(\cdot) \end{bmatrix} \end{cases}$$

where $f_3(\cdot)$ is given by Eq. (3.70), which belongs to sector [0, $\frac{s+\rho+1}{\rho+0.5}$].

A Lur'e representation for the circuit that generates the hyperchaotic n-scroll attractor is given by

$$B = \begin{bmatrix} \alpha_1\Delta m & \alpha_1(m_2 - m_1) & \cdots & \alpha_1(m_{2n_h-1} - m_{2n_h-2}) \\ 0 & 0 & \cdots & 0 \\ -\beta\Delta m & -\beta(m_2 - m_1) & \cdots & -\beta(m_{2n_h-1} - m_{2n_h-2}) \\ 0 & 0 & \cdots & 0 \end{bmatrix}$$

$$A = \begin{bmatrix} -\alpha_1 m_0 & \alpha_1 m_0 & -\alpha_2 & 0 \\ \beta m_0 & -\beta m_0 & 0 & -\beta \\ 1 & 0 & 1 & 0 \\ 0 & \gamma & 0 & 0 \end{bmatrix}, C = \begin{bmatrix} -1 & 1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ . & . & . & . \\ . & . & . & . \\ -1 & 1 & 0 & 0 \end{bmatrix}. \quad (3.91)$$

The vector valued nonlinearity $\sigma = [\sigma_1 \ \sigma_2 \cdots \sigma_{2q-1}]^T$ with $\sigma_i(x; c_i) = \frac{1}{2}(|x + c_i| - |x - c_i|)$ where $i = 1, 2, \ldots, n_h = 2q - 1$ belong to sector $[0, 1]$.

Also n-scroll hypercube attractors from the 1-D CNN with unidirectional coupling and diffusive coupling between cells are representable in Lur'e form;

$$B = \begin{bmatrix} B_c & 0_{3\times1} & \cdots & 0_{3\times1} \\ 0_{3\times1} & B_c & \cdots & 0_{3\times1} \\ . & . & \cdots & . \\ . & . & \cdots & . \\ . & . & \cdots & . \\ 0_{3\times1} & 0_{3\times1} & \cdots & B_c \end{bmatrix}, C = \begin{bmatrix} C_c & 0_{1\times3} & \cdots & 0_{1\times3} \\ 0_{1\times3} & C_c & \cdots & 0_{1\times3} \\ . & . & \cdots & . \\ . & . & \cdots & . \\ . & . & \cdots & . \\ 0_{1\times3} & 0_{1\times3} & \cdots & C_c \end{bmatrix}, \quad (3.92)$$

$$A = \begin{bmatrix} A_c & 0_{3\times3} & 0_{3\times3} & \cdots & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ -T & A_c + T & 0_{3\times3} & \cdots & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & -T & A_c + T & \cdots & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & \cdots & 0_{3\times3} & -T & A_c + T \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & \cdots & 0_{3\times3} & 0_{3\times3} & A_c \end{bmatrix} \quad (3.93)$$

and

$$A = \begin{bmatrix} A_D & D & 0_{3\times3} & \cdots & 0_{3\times3} & 0_{3\times3} & D \\ D & A_D & D & \cdots & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & \cdots & D & A_D & D \\ D & 0_{3\times3} & 0_{3\times3} & \cdots & 0_{3\times3} & D & A_D \end{bmatrix} \quad (3.94)$$

where $A_D = A_c - 2D$, respectively. A_c, B_c and C_c are equal to A, B and C in Eq. (3.88), respectively. $0_{n\times m}$ is a n-by-m zero matrix and

$$T = \begin{bmatrix} 0 & 0 & 0 \\ 0 & K & 0 \\ 0 & 0 & 0 \end{bmatrix}, D = \begin{bmatrix} D_x & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

The vector valued nonlinearity $\sigma = [\Sigma^{(1)} \ \Sigma^{(2)} \ \cdots \ \Sigma^{(L)}]^T$ where $\Sigma^{(j)} = [\sigma_1 \ \sigma_2 \cdots \sigma_{2q-1}]^T$ with $\sigma_i(x; c_i) = \frac{1}{2}(|x + c_i| - |x - c_i|)$ for $i = 1, 2, \cdots, n_h = 2q - 1$ belongs to sector $[0, 1]$.

3.7 Conclusion

As stated in [110], the implementation and characterization of several electronic circuits exhibiting chaotic behavior was the first historical achievement which brought complex nonlinear phenomena and chaos from mathematical abstraction to the core of electrical engineering. The relative simplicity of Chua's circuit provides a convenient model for the dynamics and bifurcation phenomena in electronic circuits. Therefore, Chua's circuit has become a paradigm for chaos. The generalization of Chua's circuit in the 90s was a challenge in this achievement which led to the discovery of new strange attractors in Chua's circuit by systematically modifying the nonlinearity. Today, these attractors have been called multi-scroll attractors. In this chapter, we have given an overview of these developments and faced the reader with further generalizations of multi-scroll chaotic attractors, families of scroll grid attractor. Until the introduction of grid attractors, it was not possible to generate the scrolls in different directions. We have shown that it is possible to create strange attractors whose scrolls can be located in any state variable direction. Following these ideas, the design of new attractors depends on the designer's imagination, as the presented scroll grid attractors are just instance derived from the new proposed family which might be further extended in the future. The most recent achievement in families of scroll grid attractor has been presented by Han *et al.* [104] who used a linear second-order system with a hysteresis series to generate 1-D and 2-D grid scroll attractors. In this chapter, we also carried our knowledge and experience on multi-scroll chaotic attractor into hyperchaotic attractors and obtained multi-scroll hyperchaotic attractors circuits. Multi-scroll chaotic and hyperchaotic attractors can be represented in Lur'e form. Hence, many results concerning stability and synchronization, which is the topic of the next Chapter, are applicable to it. Furthermore, we proposed scroll maps that imitate the behavior of a two-sided Poincaré map of n-scroll attractors. Like multi-scroll chaotic attractors, scroll maps can be systematically generated. Multi-scroll

chaotic and hyperchaotic attractors circuits are useful in many chaos-based applications. The applications of these attractors, which are studied in this thesis, is the topic of Chapter 5.

Chapter 4

Synchronization of Chaotic Lur'e Systems

In this Chapter a systematic overview is given of methods for master-slave synchronization of Lur'e systems. The Chapter is organized as follows. In Section 4.1 we discuss the synchronization problem. For the autonomous case we link master-slave synchronization with absolute stability theory. In Section 4.2, synchronization criteria for master-slave synchronization are given for both full static state feedback and the dynamic output feedback case. Robust synchronization is studied in Section 4.3 for non-identical master and slave systems and sufficient conditions for uniform synchronization with a bound are derived. Time delay (or propagation delay) between master and slave systems is considered in Section 4.4 with delay-independent and delay-dependent criteria for synchronization. For master-slave schemes with external inputs, methods of nonlinear H_∞ synchronization are explained for both full static state feedback and dynamic output feedback cases in Section 4.5. The influence of parameter mismatch between the master and slave systems is studied with respect to the method of nonlinear H_∞ synchronization in Section 4.6. In Section 4.7 we present results on impulsive synchronization. In Section 4.8 we explain how to design the controllers from the given synchronization criteria for achieving synchronization and finally in Section 4.9 a number of illustrations are given with computer simulation as well as real electronic circuit implementations. New contributions are made in Sections 4.4.1, 4.4.2, 4.9.1, 4.9.2 and 4.9.3.

4.1 Synchronization

Synchronization is encountered in various fields of science, in engineering and in social behavior. Perhaps one of the earliest and best-known observation of the synchronization phenomenon was given by the Dutch researcher Christiaan Huygens in 1665 [199]. Huygens made the observation of synchronization for two weakly coupled pendulum clocks. A recent observation of synchronization has been reported on two coupled biological neurons [74, 273]. Since the early work by Pecora and Carroll [197] on synchronization phenomena of chaotic systems, the research on synchronization has moved towards chaotic systems. This was a challenge in synchronization because it is well-known that chaotic systems are extremely sensitive with respect to initial conditions. Over the last decade, several synchronization schemes have been proposed and the theoretical and experimental analysis of these schemes has been collected in the monographs [199, 29], edited books [28, 30, 31] and special issues [133, 187, 19].

The coupling configuration is the basic feature that helps to distinguish between different schemes and the process leading to synchronized states. Two main cases are unidirectional coupling and bidirectional coupling. The coupling is called unidirectional coupling when the coupling is considered in one single direction, either for identical systems

$$\mathcal{M}: \ \dot{x} = f(x,t) \tag{4.1}$$
$$\mathcal{S}: \ \dot{z} = f(z,t) + u(x,z) \tag{4.2}$$

with master system $\mathcal{M}$ and slave system $\mathcal{S}$; or for non-identical systems

$$\mathcal{S}: \ \dot{z} = g(z,t) + u(x,z) \tag{4.3}$$

where state vectors $x, z \in \mathbb{R}^n$, $u(\cdot) : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$, $f(\cdot) : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$ and $g(\cdot) : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$. The Pecora and Carroll method (*drive-response system*) [197], the master-slave method [286], the negative feedback [122] and the active-passive decomposition method [196] are examples of this scheme. On the other hands, the coupling is called bidirectional when both systems (either identical or non-identical) are connected in such a way that they mutually influence each other

$$\begin{aligned} \mathcal{M}_1: \ \dot{x} &= f(x,t) + u_2(x,z) \\ \mathcal{M}_2: \ \dot{z} &= f(z,t) + u_1(x,z) \end{aligned} \tag{4.4}$$

where $u_1(\cdot), u_2(\cdot) : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$. This Chapter is devoted to the master-slave synchronization scheme. This implies that the master system (4.1) evolves

freely and drives the dynamics of the slave system (4.2). Hence, the slave system follows the dynamics of the master system.

Moreover many different synchronization schemes have been studied under both coupling configurations : complete synchronization [197, 42, 139, 24, 286, 247, 64, 188], robust synchronization [286, 241, 242, 142], generalized synchronization [141, 249], partial synchronization [109], nonlinear H_∞ synchronization [250, 243], phase and lag synchronization [19] and impulsive synchronization [315, 254, 255, 256]. The pioneering work of Pecora and Carroll in [197] is an example of complete synchronization. The criteria for this synchronization method were provided by computation of conditional Lyapunov exponents [238, 195] and it is still one of the main tools in synchronization studies.

The large interest in synchronization and its potential application to the field of communications was brought to the attention of the systems and control engineering community for example in [4, 133, 187]. As a positive result of the dialog between the systems and control engineering and the chaos researchers community, methods for analysis and design of synchronization schemes have been developed in a more systematic way and for different classes of nonlinear systems. The link was made between synchronization and absolute stability theory of Lur'e systems [240, 253] and with observer design in [181, 189].

This thesis focuses on the class of Lur'e system. We have already shown in Chapter 3 that several chaotic and hyperchaotic systems including Chua circuit [38], n-scroll circuits [244], coupled Chua circuits [252] and others are representable in Lur'e form.

Several synchronization criteria have been developed for master-slave schemes in the context of Lur'e systems. The rigorous mathematical proofs of these criteria have been obtained from global asymptotic stability analysis of the error systems with error variable $e = x - z$. This is in contrast with other methods of providing synchronization criteria which require the computation of numerical quantities such as conditional Lyapunov exponents. Here, we discuss master-slave synchronization methods for the class of Lur'e systems. Synchronization problems for different methods are expressed as Linear Matrix Inequalities [22]. Figure 4.1 shows the synchronization schemes which are studied in this Chapter together with design purposes.

Absolute stability

By synchronization one usually means that $\|x - z\|_2 \to 0$ as $t \to \infty$ where x and z denote the state vectors of the master and slave systems, respectively. Many results on synchronization in the literature are about local synchroniza-

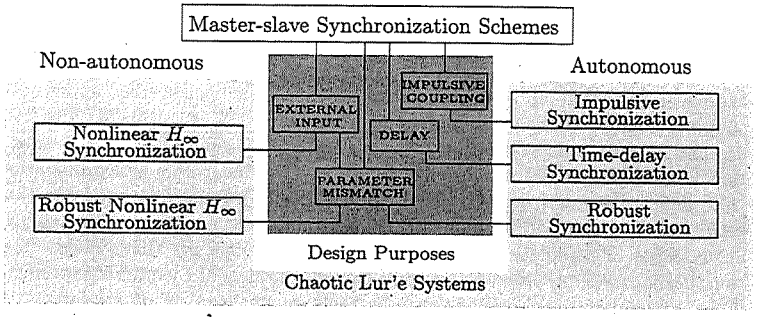


Figure 4.1: Different synchronization schemes for the class of Lur'e systems and their design purposes.

tion. However, for Lur'e systems (see Section 3.6) it is straightforward to derive conditions for global synchronization in the sense that the synchronization property holds for all possible initial conditions $(x(0), z(0))$.

The study of absolute stability, which is also known as the *Lur'e problem*, dates back from classical nonlinear control theory with fundamental contributions such as the circle criterion and the Popov criterion, corresponding to a quadratic Lyapunov function

$$V(x) = x^T P x, \quad P = P^T > 0 \tag{4.5}$$

and a Lur'e-Postnikov Lyapunov function

$$V(x) = x^T P x + \mu \int_0^x \sigma(\gamma) d\gamma; \quad P = P^T > 0, \mu \geq 0, \tag{4.6}$$

respectively. The criteria can be either expressed in terms of matrix inequalities [22] or frequency domain criteria by applying the well-known Kalman-Yacubovich-Popov (KYP) Lemma [134, 275]. The problem of interest here is to study the stability of the origin, but for a class of nonlinearities that satisfies a given sector condition as in Definition 3.1. If the origin is globally and asymptotically stable for all nonlinearities belonging to the sector, the system is said to be absolutely stable.

4.2 Master-slave synchronization: autonomous case

For the class of Lur'e systems, the synchronization problem was first explicitly linked to the absolute stability problem of Lur'e systems by Suykens and Vandewalle in [247]. Then Curran and Chua [63] evaluated synchronization criteria given by Pecora and Carroll [197], Cuomo and Oppenheim [62] and Wu and Chua [287] in the context of absolute stability theory. Furthermore while a quadratic Lyapunov function (4.5) was employed for the synchronization by Curran and Chua [63], Lur'e-Postnikov Lyapunov function (4.6) was employed by Suykens and Vandewalle [247]. Later, Curran, Suykens and Chua [64] showed how the more general Lur'e-Postnikov Lyapunov function of absolute stability theory can be employed in the development of synchronization criteria for master-slave synchronization scheme.

In this chapter, we will present the systematic application of quadratic Lyapunov functions for the development of synchronization criteria. Also for different synchronization schemes such as robust synchronization, nonlinear H_∞ synchronization, we will use quadratic Lyapunov functions in order to obtain synchronization criteria.

4.2.1 Full static state error feedback

For master-slave synchronization of identical Lur'e systems one has the scheme

$$\begin{aligned} \mathcal{M}: & \left\{ \begin{array}{rcl} \dot{x} & = & Ax + B\,\sigma(Cx) \\ p & = & Hx \end{array} \right. \\ \mathcal{S}: & \left\{ \begin{array}{rcl} \dot{z} & = & Az + B\,\sigma(Cz) + u \\ q & = & Hz \end{array} \right. \\ \mathcal{C}: & \quad u \;=\; F(p-q) \end{aligned} \tag{4.7}$$

with master system $\mathcal{M}$, slave system $\mathcal{S}$ and full static state error feedback controller $\mathcal{C}$. The master and slave system contain two identical Lur'e systems, i.e. $\mathcal{M}$ and $\mathcal{S}$, with the state vectors $x, z \in \mathbb{R}^n$, system matrices $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times n_h}$, $C \in \mathbb{R}^{n_h\times n}$ and nonlinearity $\sigma(\cdot) : \mathbb{R}^{n_h} \to \mathbb{R}^{n_h}$ which are considered as described for the system (3.85). At the master and slave system, a linear transformation $H \in \mathbb{R}^{n\times n}$ is applied to the state vectors. The subsystems have output vectors $p, q \in \mathbb{R}^n$ and $u \in \mathbb{R}^n$ for the slave system. Full static state error feedback between the output q of $\mathcal{S}$ and the output p of $\mathcal{M}$ is applied with feedback matrix $F \in \mathbb{R}^{n\times n}$ at the slave system.

Investigating synchronization of the schemes is done by stability analysis of the error system

$$\mathcal{E}: \quad \dot{e} = (A-F)e + B\,\eta(Ce;z) \tag{4.8}$$

where $\eta(Ce;z) = \sigma(Ce+Cz) - \sigma(Cz)$ and $e = x - z$.

In order to derive criteria for synchronization of scheme (4.7), we first have to introduce an assumption on the nonlinearity $\eta(Ce;z)$ in the error system (4.8).

Assumption 4.1 *The nonlinearity $\eta(Ce;z)$ belongs to sector $[0,k]$:*

$$0 \le \frac{\eta_i(c_i^T e;z)}{c_i^T e} = \frac{\sigma_i(c_i^T e + c_i^T z) - \sigma_i(c_i^T z)}{c_i^T e} \le k, \quad \forall e, z \tag{4.9}$$

where $i = 1, 2, ..., n_h$ and c_i^T denotes the i^{th} row vector of C.

The following inequality then holds [22, 134, 275]:

$$\eta_i(c_i^T e;z)[\eta(c_i^T e;z) - k c_i^T e] \le 0, \;\; \forall e, z; \;\; i = 1, 2..., n_h. \tag{4.10}$$

Therefore, the error system is also a Lur'e system. Hence, the problem of synchronization of Lur'e systems can be reduced to the problem of absolute stability of the error system [134, 275].

Theorem 4.1 (Master-slave synchronization using full static state error feedback) [64, 63] *Suppose Assumption 4.1 holds and that there exist matrices $0 < P = P^T \in \mathbb{R}^{n\times n}$, $\Lambda = \mathrm{diag}(\lambda_i)$ with $\lambda_i \ge 0$ for $i = 1, ..., n_h$ and*

$$Z = \begin{bmatrix} (A-FH)^T P + P(A-FH) & PB + kC^T\Lambda \\ B^T P + k\Lambda C & -2\Lambda \end{bmatrix} < 0 \tag{4.11}$$

then the master-slave system (4.7) globally and asymptotically synchronizes.

Proof: Taking the time derivative of the following quadratic and positive definite Lyapunov function

$$V(e) = e^T P e, \quad P = P^T > 0 \tag{4.12}$$

and applying the $\mathcal{S}$-procedure using the inequalities (4.10), one obtains

$$\begin{aligned} \dot{V}(e) &= \dot{e}^T P e + e^T P \dot{e} \\ &= e^T[(A-FH)^T P + P(A-FH)]e + e^T P B\eta + \eta^T B^T P e \\ &\le e^T[(A-FH)^T P + P(A-FH)]e + e^T P B\eta + \eta^T B^T P e \\ &\quad - \sum_i 2\lambda_i \eta_i(\eta_i - k c_i^T e) \end{aligned} \tag{4.13}$$

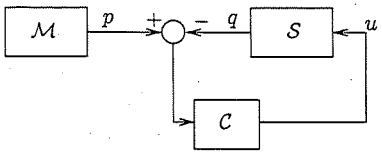


Figure 4.2: Master-slave synchronization scheme with dynamic measurement feedback.

Writing this as a quadratic form in $\zeta = [e^T \ \eta^T]^T$ one obtains

$$\dot{V}(e) \leq \zeta^T Z \zeta < 0. \tag{4.14}$$

This expression is negative $\forall x, z \in \mathbb{R}^n$ if Z is negative definite. □

4.2.2 Dynamic output feedback

In the scheme (4.7) it is assumed that the full state vectors x, z can be measured. But, measuring all the state variables of the system is inconvenient or often even impossible. Therefore, Suykens *et al.* [240] raised the question of how synchronization could be obtained for a general class of nonlinear system based on e.g. a single or fewer measurement signals and fewer control signals. An alternative scheme with linear dynamic output feedback $\mathcal{C}$ (Figure 4.2) [240] has been proposed

$$\begin{aligned}
\mathcal{M}: &\begin{cases} \dot{x} &= Ax + B\,\sigma(Cx) \\ p &= Hx \end{cases} \\
\mathcal{S}: &\begin{cases} \dot{z} &= Az + B\,\sigma(Cz) + Fu \\ q &= Hz \end{cases} \\
\mathcal{C}: &\begin{cases} \dot{\rho} &= E\rho + G(p-q) \\ u &= M\rho + N(p-q) \end{cases}
\end{aligned} \tag{4.15}$$

with $\rho \in \mathbb{R}^{n_c}$, outputs $p, q \in \mathbb{R}^l$, control input $u \in \mathbb{R}^{n_u}$ where $l, n_u \leq n$. For the master and slave system, we consider the state vectors, system matrices and nonlinearity as described for the system (3.85). A linear transformation $H \in \mathbb{R}^{l \times n}$ is applied to the state vectors x and z in order to obtain the outputs of the master and slave system, respectively. The difference between the outputs of the master and the slave system, i.e. p and q, is taken here as an input of the linear dynamic controller with system matrices $E \in \mathbb{R}^{n_c \times n_c}$, $G \in \mathbb{R}^{n_c \times l}$, $M \in \mathbb{R}^{n_u \times n_c}$ and $N \in \mathbb{R}^{n_u \times l}$. The slave system is controlled by means of the

control signal u through the matrix $F \in \mathbb{R}^{n \times n_u}$. The error system becomes

$$\mathcal{E}: \begin{cases} \dot{e} &= (A - FN)e - FM\rho + \eta(Ce; z) \\ \dot{\rho} &= GHe + E\rho \end{cases} \tag{4.16}$$

where $e = x - z$. Defining the state vector $\xi = [e^T \ \rho^T]^T$ and taking a positive definite quadratic Lyapunov function (4.12) with

$$P = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}, \ P = P^T > 0 \tag{4.17}$$

a sufficient condition for global asymptotic stability of Eq. (4.16) is derived in the following Theorem.

Theorem 4.2 (Master-slave synchronization using dynamic output feedback) [240] *Suppose Assumption 4.1 holds and that there exist matrices* $0 < P = P^T \in \mathbb{R}^{n+n_c \times n+n_c}$ *(Eq. (4.17)) and* $\Lambda = \mathrm{diag}(\lambda_i)$ *with* $\lambda_i \geq 0$ *for* $i = 1, ..., n_h$ *and*

$$Y = Y^T = \begin{bmatrix} Y_{11} & Y_{12} & Y_{13} \\ . & Y_{22} & P_{21}B \\ . & . & -2\Lambda \end{bmatrix} < 0 \tag{4.18}$$

with

$$\begin{aligned}
Y_{11} &= (A - FNH)^T P_{11} + P_{11}(A - FNH) + H^T G^T P_{21} + P_{12} GH, \\
Y_{12} &= (A - FNH)^T P_{12} + H^T G^T P_{22} - P_{11} FM + P_{12} E, \\
Y_{13} &= P_{11} B + kC^T \Lambda, \\
Y_{22} &= E^T P_{22} + P_{22} E - M^T F^T P_{12} - P_{21} FM
\end{aligned}$$

then, the master-slave system (4.15) synchronizes.

Proof: Taking the time derivative of the Lyapunov function (4.5) and applying the $\mathcal{S}$-procedure using the inequalities of Eq. (4.10), one obtains:

$$\begin{aligned}
\dot{V}(\xi) &= \dot{\xi}^T P \xi + \xi^T P \dot{\xi} \\
&\leq \dot{\xi}^T P \xi + \xi^T P \dot{\xi} - \sum_i 2\lambda_i \eta_i (\eta_i - k c_i^T e) \\
&= [(A - FNH)e - FM\rho + B\eta]^T (P_{11} e + P_{12}\rho) \\
&\quad + (E\rho + GHe)^T (P_{21} e + P_{22}\rho) \\
&\quad + (e^T P_{11} + \rho^T P_{12})[(A - FNH)e - FM\rho + B\eta] \\
&\quad + (e^T P_{12} + \rho^T P_{22})(E\rho + GHe) - 2\eta^T \Lambda(\eta - kCe)
\end{aligned} \tag{4.19}$$

Writing this as a quadratic form in $\zeta = [e^T \ \rho^T \ \eta^T]^T$ one obtains

$$\dot{V} \leq \zeta^T Y \zeta < 0. \tag{4.20}$$

This expression is negative for all non-zero ζ provided Y is negative definite. □

4.3 Robust synchronization

So far, we have only considered master-slave systems with two identical Lur'e systems. Now, we study the influence of parameter mismatch between the Lur'e systems. Such issues of robustness are important because non-robust schemes may cause loss of synchronization. This issue was first addressed by Wu and Chua [286] who defined a synchronization error bound because zero synchronization error cannot be achieved. In [241], Suykens *et al.* obtained a sufficient conditions for uniform synchronization with a bound based on a quadratic Lyapunov function. Here, we present the result of Suykens *et al.* [241].

4.3.1 Full static state error feedback

The case of non-identical Lur'e systems is described by

$$\mathcal{M}: \begin{cases} \dot{x} &= A_1 x + B_1 \sigma(C_1 x) \\ p &= Hx \end{cases}$$

$$\mathcal{S}: \begin{cases} \dot{z} &= A_2 z + B_2 \sigma(C_2 z) + u \\ q &= Hz \end{cases} \tag{4.21}$$

$$\mathcal{C}: \quad u = F(p-q)$$

with master system $\mathcal{M}$, slave system $\mathcal{S}$ and full static state error feedback controller $\mathcal{C}$. For the master and slave system, we consider the state vectors and nonlinearity as described for the system (4.7) in Section 4.2.1. The system matrices are $A_1, A_2 \in \mathbb{R}^{n \times n}$, $B_1, B_2 \in \mathbb{R}^{n \times n_h}$ and $C_1, C_2 \in \mathbb{R}^{n_h \times n}$. The parameter mismatch between the system, $\|A_1 - A_2\|_2$, $\|B_1 - B_2\|_2$ and $\|C_1 - C_2\|_2$ is assumed to be relatively small.

The error system can be decomposed into two terms, according to [286]:

$$\mathcal{E}: \quad \dot{e} = \varrho(x,z) + \varphi(x) \tag{4.22}$$

with

$$\varrho(x,z) = (A_2 - F)e + B_2 \eta(C_2 e; z) \tag{4.23}$$

$$\varphi(x) = \Delta A x + B_1 \sigma(C_1 x) - B_2 \sigma(C_2 x) \tag{4.24}$$

where $e = x - z$ and $\Delta A = A_1 - A_2$. The first term $\varrho(x,z)$ is in the form of the error system (4.8) of master-slave system in Section 4.2.1, while the second term $\varphi(x)$ contains the influence of the parameter mismatch. The aim is here to design the feedback matrix F such that the scheme is robust with respect to the parameter mismatch.

In order to derive criteria for synchronization of the scheme (4.21), the following additional assumption on the norm of the state vector of the master system has to be made.

Assumption 4.2 *Master system $\mathcal{M}$ satisfies the condition that there exists a positive real constant δ such that, for any initial condition x_0, there exists a time instant $T(x_0)$ for which*

$$\|x(t; x_0)\|_2 \leq \delta, \quad \forall t \geq T. \tag{4.25}$$

This assumption is based on the work of Wu and Chua [286]. From a practical point of view this is a reasonable assumption, because we are interested in employing a master system that is in a chaotic regime.

Because the master and slave system are non-identical the synchronization error will not tend asymptotically to zero. Therefore the following definition of synchronization with error bound ϵ is employed [286].

Definition 4.1 (Uniform Synchronization) *The synchronization scheme (4.21) uniformly synchronizes with error bound ϵ if there exists a constant $\delta_0 > 0$ and a time instant $T \geq 0$ such that if*

$$\|x(0) - z(0)\|_2 \leq \delta_0 \tag{4.26}$$

then

$$\|x(t) - z(t)\|_2 \leq \epsilon \tag{4.27}$$

for all $t \geq T$.

Theorem 4.3 (Robust master-slave synchronization using full static state error feedback) [241] *Suppose Assumptions 4.1 and 4.2 hold and that*

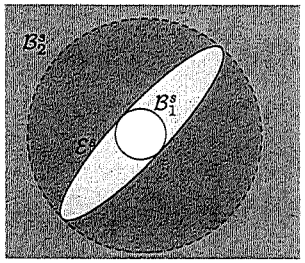


Figure 4.3: Illustration of the balls $\mathcal{B}_1^s$, $\mathcal{B}_2^s$ and the ellipsoid $\mathcal{E}^s$.

there exists $0 < P = P^T \in \mathbb{R}^{n\times n}$, $\Lambda = \mathrm{diag}(\lambda_i)$ *with* $\lambda_i \geq 0$ *for* $i = 1, ..., n_h$ *and*

$$Y = Y^T = \begin{bmatrix} \Upsilon & PB_2 + kC_2^T\Lambda \\ B_2^TP + k\Lambda C & -2\Lambda \end{bmatrix} < 0 \tag{4.28}$$

where $\Upsilon = (A_2 - FH)^TP + P(A_2 - FH) + \alpha I$. *Define* $\gamma = 2\beta\delta\sigma_{max}(P)$, *where* β *is defined*

$$\|\Delta Ax + B_1\sigma(C_1x) - B_2\sigma(C_2x)\|_2 \leq \beta\|x\|_2, \quad \forall x \in \mathbb{R}^n \tag{4.29}$$

and $\sigma_{max}(P)$ *denotes the maximal singular value of* P. *Then the synchronization scheme (4.21) is uniformly synchronizing with error bound* $\sqrt{c_2}$, *if there exist positive real constants* c *and* c_2 *such that*

$$\mathcal{B}_1^s = \{e|e^Te \leq \frac{\gamma^2}{\alpha^2}\} \subset \mathcal{E}^s = \{e|e^TPe \leq c\} \subset \mathcal{B}_2^s = \{e|e^Te \leq c_2\}. \tag{4.30}$$

Proof: Taking the time derivative of the Lyapunov function (4.5) and applying the $\mathcal{S}$-procedure using the inequalities of Eq. (4.10), one obtains:

$$\begin{aligned} \dot{V} &= \dot{e}^TPe + e^TP\dot{e} \\ &\leq [(A_2 - FH)e + B_2\eta]^TPe + e^TP[(A_2 - FH)e + B_2\eta] \\ &\quad + 2e^TP[\Delta Ax + B_1\sigma(C_1x) - B_2\sigma(C_2x)] - \sum_i 2\lambda_i\eta_i(\eta_i - kc_i^Te). \end{aligned} \tag{4.31}$$

Using the inequalities of Eqs. (4.25) and (4.29), we obtain

$$\dot{V} \leq \zeta^TY_0\zeta + 2\beta\delta\sigma_{max}(P)\|e\|_2 \tag{4.32}$$

where $\zeta = [e^T \ \eta^T]^T$ and

$$Y_0 = \begin{bmatrix} (A_2 - FH)^TP + P(A_2 - FH) & PB_2 + kC_2^T\Lambda \\ B_2^TP + k\Lambda C & -2\Lambda \end{bmatrix} < 0. \tag{4.33}$$

By imposing the upper bound $\zeta^TY_0\zeta \leq -\alpha e^Te$, the matrix inequality (4.28) is obtained. This yields

$$\begin{aligned} \dot{V} &\leq -\alpha\|e\|_2^2 + \gamma\|e\|_2 \\ &= [-\alpha\|e\|_2 + \gamma]\|e\|_2. \end{aligned} \tag{4.34}$$

The time derivative of the Lyapunov function (4.5) is negative if

$$\|e(t)\| > \frac{\gamma}{\alpha}, \quad \forall t. \tag{4.35}$$

$\mathcal{E}^s = \{e|e^TPe \leq c\}$ is the smallest ellipsoid containing the ball $\mathcal{B}_1^s$. The ellipsoid $\mathcal{E}^s$ is directly related to the Lyapunov function (4.5). Hence the limit set of error system is nonempty, closed, bounded and belongs to $\mathcal{E}^s$ for every initial condition. Therefore, the error system will uniformly synchronize with error bound $\sqrt{c_2}$ where c_2 is determined such that $\mathcal{B}_2^s$ is the smallest ball containing the ellipsoid $\mathcal{E}^s$. □

In order to construct $\mathcal{B}_2^s$ and $\mathcal{E}^s$ (also to find c_2 and c) for given $\mathcal{B}_1^s$, Suykens *et al.* [241] have used an ellipsoidal approximation method based on convex optimization procedures for finding an outer approximation of the union of ellipsoids and an intersection of ellipsoids [22]. Hence, the error bound is given by $\sqrt{c_2} = \frac{\gamma}{\alpha}\kappa(P)$ with $\kappa = \frac{\sigma_{max}(P)}{\sigma_{min}(P)}$ from Figure 4.3.

4.3.2 Dynamic output feedback

The robust synthesis of dynamic output feedback for master-slave synchronization of Lur'e systems was also presented by Suykens *et al.* in [241]. The case of non-identical master-slave system with linear dynamic output feedback $\mathcal{C}$ is described by

$$\begin{aligned} \mathcal{M}: &\begin{cases} \dot{x} &= A_1x + B_1\,\sigma(C_1x) \\ p &= Hx \end{cases} \\ \mathcal{S}: &\begin{cases} \dot{z} &= A_2z + B_2\,\sigma(C_2z) + Fu \\ q &= Hz \end{cases} \\ \mathcal{C}: &\begin{cases} \dot{\rho} &= E\rho + G(p - q) \\ u &= M\rho + N(p - q) \end{cases} \end{aligned} \tag{4.36}$$

with master system $\mathcal{M}$, slave system $\mathcal{S}$ and dynamic output feedback controller $\mathcal{C}$. For the master, slave and controller systems, we consider the state vectors, systems matrices and nonlinearity as described for the system (4.15) in Section 4.2.2. The system matrices A_i, B_i and C_i are defined in previous section for $i = 1, 2$.

One obtains the error system:

$$\mathcal{E}: \begin{cases} \dot{e} = (A_2 - FNH)e + B_2\,\eta(C_2 e; z) \\ \qquad + \Delta A x + B_1\sigma(C_1 x) - B_2\sigma(C_2 x) - FM\rho \\ \dot{\rho} = GHe + E\rho \end{cases} \tag{4.37}$$

where $e = x - z$. Defining the state vector $\xi = [e^T \ \rho^T]^T$ and taking a positive definite quadratic Lyapunov function $V(\xi)$ (4.5) with Eq. (4.17), a sufficient condition for uniform synchronization of scheme (4.36) is derived in following Theorem.

Theorem 4.4 (Robust master-slave synchronization with linear dynamic output feedback) [241] *Suppose Assumption 4.1 and 4.2 hold and that there exists* $0 < P = P^T \in \mathbb{R}^{n+n_c \times n+n_c}$, $\Lambda = \mathrm{diag}(\lambda_i)$ *with* $\lambda_i \geq 0$ *for* $i = 1, ..., n_h$, *positive real constants* α_1, α_2 *and*

$$Z = Z^T = \begin{bmatrix} Z_{11} + \alpha_1 I & Z_{12} & Z_{13} \\ Z_{12}^T & Z_{22} + \alpha_2 I & Z_{23} \\ Z_{13}^T & Z_{23}^T & -2\Lambda \end{bmatrix} < 0 \tag{4.38}$$

with

$$\begin{aligned} Z_{11} &= (A_2 - FNH)^T P_{11} + P_{11}(A_2 - FNH) + H^T G^T P_{21} + P_{12} GH, \\ Z_{12} &= (A_2 - FNH)^T P_{12} + H^T G^T P_{22} - P_{11} FM + P_{12} E, \\ Z_{13} &= P_{11} B_2 + k C_2^T \Lambda, \\ Z_{22} &= E^T P_{22} + P_{22} E - M^T F^T P_{12} - P_{21} FM, \\ Z_{23} &= P_{21} B_2. \end{aligned}$$

Define $P_I = \mathrm{diag}\{I_n/\eta_1^2, I_{n_c}/\eta_2^2\}$ *with* $\eta_1 = \frac{\tau_1}{2\alpha_1} + \sqrt{\frac{\tau_1^2}{\alpha_1^2} + \frac{\tau_2^2}{\alpha_1\alpha_2}}$, $\eta_2 = \frac{\tau_2}{2\alpha_2} + \sqrt{\frac{\tau_2^2}{\alpha_2^2} + \frac{\tau_1^2}{\alpha_1\alpha_2}}$ *where* $\tau_1 = 2\beta\delta\sigma_{max}(P_{11})$, $\tau_2 = 2\beta\delta\sigma_{max}(P_{12})$ *where* β *and* δ *are defined in Eqs. (4.25) and (4.29). Then the synchronization scheme (4.21) is uniformly synchronizing with error bound* $\sqrt{c_3}$, *if there exist positive real constant* c_1, c_2, c_3 *and* c_4 *such that*

$$\mathcal{B}_1^d = \{e | e^T e \leq \eta_1^2\} \subset \mathcal{E}_1^d = \{\xi | \xi^T P_1 \xi \leq 1\},\ \mathcal{B}_2^d = \{\rho | \rho^T \rho \leq \eta_2^2\} \subset \mathcal{B}_1^d, \tag{4.39}$$

$$\mathcal{B}_3^d = \{e | e^T e \leq c_3\} \subset \mathcal{E}_3^d = \{\xi | \xi^T P_I \xi \leq c_2\},\ \mathcal{B}_4^d = \{\rho | \rho^T \rho \leq c_4\} \subset \mathcal{B}_3^d, \tag{4.40}$$

and

$$\mathcal{E}_1^d \subset \mathcal{E}_2^d = \{\xi | \xi^T \xi \leq c_4\} \subset \mathcal{E}_3^d. \tag{4.41}$$

Proof: By taking the quadratic Lyapunov function (4.5), a similar procedure as in the previous proof and imposing $\dot{V} \leq [e^T \ \rho^T \ \eta^T] Z [e^T \ \rho^T \ \eta^T]^T$ [241]. □

4.4 Time-delay synchronization scheme

In this section, we deal with propagation delay in the master-slave synchronization scheme. This problem has been recently reported in [32] which introduces the possibility of applying chaotic synchronization for optical communications. Chen and Li called this problem a phase sensitivity due to the distance between two remote chaotic systems and has reported that the existence of a time-delay can destroy the synchronization. Furthermore, the synchronization and bifurcation phenomena of two chaotic circuits which are coupled by a delay line have been investigated e.g. by Koike *et al.* [145]. Experimental confirmation of synchronization of two chaotic circuits with the existence of a delay has been given by Kawate *et al.* [127]. A theoretical study of this problem has been done by Yalçın *et al.* [305].

On the other hand, in the area of control theory time-delay systems have been investigated and it is well-known that delays often result into instability. Therefore, stability analysis of time delay systems is an important subject in control theory [102, 184, 119, 120, 265]. In the literature, stability criteria for time delay systems are classified into two main categories: *delay-independent criteria* [119, 120, 33] and *delay-dependent criteria* [182, 183, 34]. Recently, time-delay systems have been intensively studied in [170]. A Lyapunov-Krasovskii function [154] is a candidate function for asymptotical stability analysis of linear time-delay systems. Boyd *et al.* [22] derive linear matrix inequalities for multiple delays from a Lyapunov-Krasovskii function.

In this section, we consider master-slave schemes with identical Lur'e systems. We suppose that the output of the master system is received at the slave system with delay (τ) which is assumed to be a given value. The system is visualized in Figure 4.4. Consider the following master-slave synchronization scheme with static error feedback and time-delay τ

$$\mathcal{M}: \begin{cases} \dot{x}(t) &= Ax(t) + B\sigma(Cx(t)) \\ p(t) &= Hx(t) \end{cases}$$

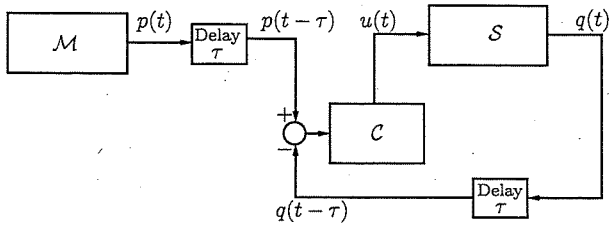


Figure 4.4: Time-delay synchronization scheme.

$$\mathcal{S}: \begin{cases} \dot{z}(t) &= Az(t) + B\sigma(Cz(t)) + u(t) \\ q(t) &= Hz(t) \end{cases} \tag{4.42}$$

$$\mathcal{C}: \quad u(t) = G(p(t-\tau) - q(t-\tau))$$

with master system $\mathcal{M}$, slave system $\mathcal{S}$ and controller $\mathcal{C}$ (Figure 4.4). Master and slave systems are Lur'e systems and we consider the state vectors, system matrices and nonlinearities as described for the system (3.85). At the master and slave system, a linear transformation $H \in \mathbb{R}^{l\times n}$ $(l \leq n)$ is applied to the state vectors. The subsystems have output vectors $p, q \in \mathbb{R}^l$. The scheme aims at synchronizing the master system to the slave system by applying full state error feedback to the slave system using a control signal $u \in \mathbb{R}^n$ with feedback matrix $G \in \mathbb{R}^{n\times l}$. Master-slave synchronization has been proved for $\tau = 0$ in Section 4.2. The difference between Section 4.2 and this section is the time-delay in the outputs i.e. $p(t-\tau)$ and $q(t-\tau)$ instead of $p(t)$ and $q(t)$, respectively.

4.4.1 Error system of time-delay synchronization scheme

Defining the signal $e = x - y$ one obtains the error system $(\mathcal{E})$:

$$\mathcal{E}: \dot{e} = Ae + B\eta(Ce; y) + Fe(t-\tau) \tag{4.43}$$

with $F = -GH$.

Stability of the error system without time-delay $(\tau = 0)$ in the feedback has been derived for quadratic Lyapunov functions (4.5) in Section 4.2. The Lyapunov-Krasovskii function is a candidate Lyapunov function for time-delay

systems [154, 170]

$$V_1(e) = e^T Pe + \int_{-\tau}^{0} e(t+s)^T Qe(t+s)ds, \quad P = P^T > 0,\, Q = Q^T > 0. \tag{4.44}$$

By taking a Lyapunov-Krasovskii function (4.44), it is straightforward to find a sufficient condition for global asymptotic stability of the error system $\mathcal{E}$.

Theorem 4.5 (Delay independent synchronization) *Suppose Assumption 4.1 holds and that there exists $\Lambda = \mathrm{diag}\{\lambda_i\}$ be a diagonal matrix with $\lambda_i \geq 0$ for $i = 1, 2, ...n_h$, $0 < P = P^T \in \mathbb{R}^{n\times n}$ and $0 < Q = Q^T \in \mathbb{R}^{n\times n}$ then a sufficient condition for global asymptotic stability of the error system $\mathcal{E}$ based on the Lyapunov-krasovskii function (4.44), is given by the matrix inequality:*

$$Y = \begin{bmatrix} A^TP + PA + Q & PB + kC^T\Lambda & PF \\ B^TP + k\Lambda^TC & -2\Lambda & 0 \\ F^TP & 0 & -Q \end{bmatrix} < 0. \tag{4.45}$$

Proof: By taking the time derivative of the Lyapunov-Krasovskii function (4.44) and applying the $\mathcal{S}$-procedure [22], and using the inequalities from the nonlinearities (4.10), one obtains:

$$\begin{aligned} \dot{V}_1 &= \dot{e}^TPe + e^TP\dot{e} + e(t)^TQe(t) - e(t-\tau)^TQe(t-\tau) \\ &\leq e^T(A^TP + PA)e + \eta^TB^TPe + e^TPB\eta + e(t-\tau)^TF^TPe \\ &\quad + e^TPFe(t-\tau) + e(t)^TQe(t) - e(t-\tau)^TQe(t-\tau) \\ &\quad - \sum_i 2\lambda_i\eta_i(\eta_i - kc_i^Te) \\ &= \xi^TY\xi < 0 \end{aligned} \tag{4.46}$$

where $\xi = [e(t)^T \;\; \eta^T \;\; e(t-\tau)^T]^T$. □

The matrix inequality (4.45) does not include information on the delay. Therefore this result is a delay-independent stability criterion for synchronization. A necessary condition for $Y < 0$ is that linear part of this system (A matrix) must be stable. An analysis of Chua's Circuit, 5-scroll attractors and hyperchaotic attractors shows that a feasible point can not be found for these examples with Y negative definite. In the literature it has been recommended that if delay-independent criteria fail, delay-dependent conditions should be applied [265]. The following Lyapunov-Krasovskii function is a candidate function for

a delay-dependent condition [170].

$$\begin{aligned} V_2(e) =& e^T P e + r_1 \int_{t-\tau}^{t} \int_{t+\theta}^{t} \left[e^T(s) A^T A e(s) \right] ds d\theta \\ &+ r_2 \int_{t-\tau}^{t} \int_{t-\tau+\theta}^{t} \left[e^T(s) F^T F e(s) \right] ds d\theta \end{aligned} \tag{4.47}$$

where $P = P^T > 0$ and $r_1 > 0$, $r_2 > 0$ are weighting factors. However, taking Eq. (4.47) as a candidate Lyapunov function and deriving an LMI from this would not be possible. For this reason we propose a new candidate Lyapunov function in the next section in order to find a sufficient condition for global asymptotic stability of the error system $\mathcal{E}$.

4.4.2 Delay-dependent synchronization criterion

Now we propose a new Lyapunov-Krasovskii function in order to obtain delay-dependent stability criteria:

$$\begin{aligned} V_3(e) =& e^T P e + r_1 \int_{t-\tau}^{t} \int_{t+\theta}^{t} \left[e^T(s) A^T A e(s) \right] ds d\theta \\ &+ r_2 \int_{t-\tau}^{t} \int_{t+\theta-\tau}^{t} \left[e^T(s) F^T F e(s) \right] ds d\theta \\ &+ r_3 \int_{t-\tau}^{t} \int_{t+\theta}^{t} \left[\eta^T(Ce(s); y(s)) B^T B \eta(Ce(s); y(s)) \right] ds d\theta \end{aligned} \tag{4.48}$$

where $P = P^T > 0$ and $r_1, r_2, r_3 > 0$. By taking this new Lyapunov function (4.48) we find a new sufficient condition for global asymptotic stability of the error system $\mathcal{E}$.

Theorem 4.6 (Delay dependent synchronization) *Suppose Assumption 4.1 holds and that there exists a diagonal matrix $\Lambda = \mathrm{diag}\{\lambda_i\}$ with $\lambda_i \geq 0$ for $i = 1, 2, ...n_h$, $0 < P = P^T \in \mathbb{R}^{n \times n}$ and $r_1, r_2, r_3 > 0$ and $\tau^* > 0$ be a scalar. Then a sufficient condition for global asymptotic stability of the error system (4.43) for any constant time-delay τ satisfying $0 \leq \tau \leq \tau^*$, based on the Lyapunov function (4.48), is given by the matrix inequality:*

$$Y = \begin{bmatrix} Z & PB + kC\Lambda \\ B^T P + k\Lambda C^T & r_3 \tau B^T B - 2\Lambda \end{bmatrix} < 0 \tag{4.49}$$

where $Z = P(A+F) + (A+F)^T P + r_1 \tau A^T A + r_2 \tau F^T F + (\frac{1}{r_1} + \frac{1}{r_2} + \frac{1}{r_3}) \tau P F F^T P$.

Proof: We have

$$e(t-\tau) = e(t) - \int_{-\tau}^{0} \dot{e}(t+\theta) d\theta. \tag{4.50}$$

Substituting $\dot{e}(t+\theta)$ into the Eq. (4.50) one obtains:

$$\begin{aligned} e(t-\tau) =& e(t) - \int_{-\tau}^{0} \{ Ae(t+\theta) + B\eta(Ce(t+\theta); y(t+\theta)) \\ &+ Fe(t+\theta-\tau) \} d\theta \end{aligned} \tag{4.51}$$

Substituting $e(t-\tau)$ back into the Eq. (4.43)

$$\begin{aligned} \dot{e} =& (A+F)e + B\eta(Ce; y) - F \left\{ \int_{-\tau}^{0} Ae(t+\theta) d\theta \right\} \\ &- F \left\{ \int_{-\tau}^{0} \{ B\eta(Ce(t+\theta); y(t+\theta)) + Fe(t+\theta-\tau) \} d\theta \right\} \end{aligned} \tag{4.52}$$

Taking the time derivative of the Lyapunov function Eq. (4.48) along the Eq. (4.52) we obtain

$$\begin{aligned} \dot{V}_3 =& e^T (P(A+F) + (A+F)^T P) e + e^T P B \eta(Ce; y) \\ &+ \eta^T(Ce; y) B^T P e - e^T P F \int_{-\tau}^{0} Ae(t+\theta) d\theta \\ &- e^T P F \int_{-\tau}^{0} B\eta(Ce(t+\theta); y(t+\theta)) d\theta \\ &- e^T P F \int_{-\tau}^{0} Fe(t+\theta-\tau) d\theta - \int_{-\tau}^{0} e^T(t+\theta) A^T F^T P e d\theta \\ &- \int_{-\tau}^{0} \eta^T(Ce(t+\theta); y(t+\theta)) B^T F^T P e d\theta \\ &- \int_{-\tau}^{0} e^T(t+\theta-\tau) F^T F^T P e d\theta + r_1 \tau e^T A^T A e + r_2 \tau e^T F^T F e \\ &+ r_3 \tau \eta^T(Ce; y) B^T B \eta(Ce; y) - \int_{-\tau}^{0} r_1 \left[e^T(t+\theta) A^T A e(t+\theta) \right] d\theta \\ &- \int_{-\tau}^{0} r_2 \left[e^T(t+\theta-\tau) F^T F e(t+\theta-\tau) \right] d\theta \\ &- \int_{t-\tau}^{0} r_3 \left[\eta^T(Ce(t+\theta); y(t+\theta)) B^T B \eta(e(t+\theta); y(t+\theta)) \right] d\theta. \end{aligned} \tag{4.53}$$

We have the following inequalities from [170]

$$\begin{aligned} &-\int_{-\tau}^{0} e^T PFAe(t+\theta)d\theta - \int_{-\tau}^{0} e^T(t+\theta)A^T F^T Ped\theta \\ \leq & r_1^{-1}\int_{-\tau}^{0} e^T PFF^T Ped\theta + r_1\int_{-\tau}^{0} e^T(t+\theta)A^T Ae(t+\theta)d\theta \end{aligned} \tag{4.54}$$

and

$$\begin{aligned} &-\int_{-\tau}^{0} e^T PFFe(t+\theta-\tau)d\theta - \int_{-\tau}^{0} e^T(t+\theta-\tau)F^T F^T Ped\theta \\ \leq & r_2^{-1}\int_{-\tau}^{0} e^T PFF^T Ped\theta + r_2\int_{-\tau}^{0} e^T(t+\theta-\tau)F^T Fe(t-\tau+\theta)d\theta \end{aligned} \tag{4.55}$$

and also

$$\begin{aligned} &-\int_{-\tau}^{0} e^T PFB\eta(e(t+\theta);y(t+\theta))d\theta \\ &-\int_{-\tau}^{0} \eta^T(e(t+\theta);y(t+\theta))B^T F^T Ped\theta \\ \leq & r_3^{-1}\int_{-\tau}^{0} e^T PFF^T Ped\theta \\ &+ r_3\int_{-\tau}^{0} \eta^T(e(t+\theta);y(t+\theta))B^T B\eta(e(t+\theta);y(t+\theta))d\theta. \end{aligned} \tag{4.56}$$

Using the above inequality, we get

$$\begin{aligned} \dot{V}_3(e) \leq & e^T Ze + e^T PB\eta(Ce;y) + \eta^T(Ce;y)B^T Pe \\ &+ r_3\tau\eta^T(Ce;y)B^T B\eta(Ce;y). \end{aligned} \tag{4.57}$$

Applying the $\mathcal{S}$-procedure, using inequalities from the nonlinearities (4.10), gives

$$\begin{aligned} \dot{V}_3(e) \leq & e^T Ze + e^T PB\eta(Ce;y) + \eta^T(Ce;y)B^T Pe \\ &+ r_3\tau\eta^T(Ce;y)B^T B\eta(Ce;y) + \sum_i \lambda_i\eta_i(\eta_i - kc_{*_i}{}^T e) \\ \leq & \xi^T Y\xi \end{aligned} \tag{4.58}$$

where $\xi = [e^T \ \eta^T]^T$.

If $\dot{V} < 0$ for τ^* the following inequality is satisfied for all $\tau \in [0, \tau^*]$

$$\begin{aligned} &\tau^*\{e^T(r_1 A^T A + r_2 F^T F + (\frac{1}{r_1}+\frac{1}{r_2}+\frac{1}{r_3})\Upsilon)e \\ &+ r_3\eta^T(Ce;y)B^T B\eta(Ce;y)\} \\ \leq & -e^T(P(A+F)+(A+F)^T P)e - \eta^T(Ce;y)B^T Pe - e^T PB\eta(Ce;y) \end{aligned} \tag{4.59}$$

where $\Upsilon = PFF^T P$. □

The matrix inequality (4.49) explicitly depends on the delay value. Therefore this result is a delay-dependent stability criteria for synchronization. Note that a necessary condition for synchronization here is that $A+F$ must be strictly Hurwitz.

4.4.3 Full static state feedback together with time delay

In [159], Liao and Chen also focussed on time delay synchronization issues. Their work is specifically intended to provide synchronization criteria when the master system is coupled to a slave system with a control signal depending on $p(t)$, $q(t)$ and $p(t-\tau)$, $q(t-\tau)$. The following scheme is considered by Liao and Chen [159]:

$$\begin{aligned} \mathcal{M}: & \begin{cases} \dot{x}(t) &= Ax(t) + B\sigma(Cx(t)) \\ p(t) &= Hx(t) \end{cases} \\ \mathcal{S}: & \begin{cases} \dot{z}(t) &= Az(t) + B\sigma(Cz(t)) + u(t) \\ q(t) &= Hz(t) \end{cases} \\ \mathcal{C}: & \quad u(t) = K(p(t)-q(t)) + G(p(t-\tau)-q(t-\tau)) \end{aligned} \tag{4.60}$$

with master system $\mathcal{M}$, slave system $\mathcal{S}$ and controller $\mathcal{C}$. For the scheme, we consider the state vectors, system matrices and nonlinearity as described for the system (4.42) with $K \in \mathbb{R}^{n\times n}$. The error system can be easily obtained like in the previous section

$$\mathcal{E}: \dot{e} = (A-KH)e + B\eta(Ce;y) + Fe(t-\tau) \tag{4.61}$$

which $F = -GH$.

Theorem 4.7 [159] *Suppose Assumption 4.1 holds and that there exists* $\Lambda = \mathrm{diag}\{\lambda_i\}$ *a diagonal matrix with* $\lambda_i \geq 0$ *for* $i = 1, 2, ...n_h$, $0 < P = P^T \in \mathbb{R}^{n\times n}$ *and* $0 < Q = Q^T \in \mathbb{R}^{n\times n}$. *Then a sufficient condition for global asymptotic*

stability of the error system (4.61), based on the Lyapunov-Krasovskii function (4.44), is given by the matrix inequality

$$Y = \begin{bmatrix} (A-KH)^T P + P(A-KH) + Q & PB + kC^T\Lambda & PF \\ B^T P + k\Lambda^T C & -2\Lambda & 0 \\ F^T P & 0 & -Q \end{bmatrix} < 0. \quad (4.62)$$

Proof: By taking the Lyapunov-Krasovskii function (4.44), it is straightforward to find the above condition using a similar procedure as in the proof of Theorem 4.5 and imposing $\dot{V} \leq [e(t)^T \ \eta^T \ e(t-\tau)^T]Y[e(t)^T \ \eta^T \ e(t-\tau)^T]^T$. □

Liao and Chen [159] have also further improved the results of [305] with some fairly simple algebraic sufficient conditions.

4.5 Nonlinear H_∞ synchronization: non-autonomous case

So far we have only dealt with the autonomous master-slave systems. Now we study the non-autonomous case of with master and slave systems. The nonlinear H_∞ synchronization method with a non-autonomous master-slave scheme, has been introduced in a series of papers by Suykens *et al.* [250, 243, 242] as a new approach for communication using chaos. In this approach the message signal is recovered after minimizing a so-called tracking error for the synchronization scheme. This is done by representing the scheme in standard plant form according to modern control theory [165, 21, 251]. Based on the system theoretical concept of dissipativity and supply rate with finite L_2-gain [281, 113, 290], matrix inequalities [22] are derived which characterize synchronization in terms of the input/output behavior of the scheme in this H_∞ sense. In [250, 243, 242] the cases of full state feedback, dynamic measurement feedback and problems of parameter mismatch have been investigated.

4.5.1 Full static state error feedback

In view of the nonlinear H_∞ synchronization method introduced in [250] the following scheme is considered

$$\mathcal{R}: \begin{cases} \dot{m} &= Rm + Sr \\ d &= Tm + Ur \end{cases}$$

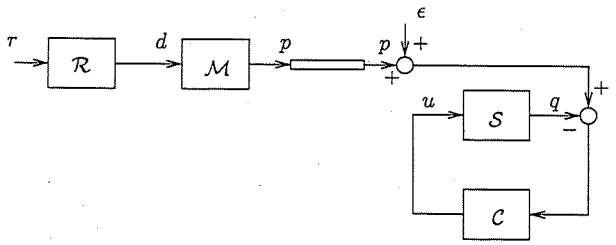


Figure 4.5: Nonlinear H_∞ synchronization scheme with master system $\mathcal{M}$ and slave system $\mathcal{S}$. The vector field of the master system is modulated by means of the signal d, which is the output of a linear filter $\mathcal{R}$ with as input the message r. The state x of $\mathcal{M}$ is linearly transformed into the transmitted signal p. The output of the channel is corrupted by the signal ϵ. The filtered message d is recovered by means of a tracting mechanism, using full static state (Section 4.5.1) or dynamic output (Section 4.5.2) error feedback controller $\mathcal{C}$. The signal q is the result of a linear state transformation (same as for $\mathcal{M}$) on the state z of $\mathcal{S}$.

$$\mathcal{M}: \begin{cases} \dot{x} &= Ax + B\,\sigma(Cx) + Dd \\ p &= Hx \end{cases} \quad (4.63)$$

$$\mathcal{S}: \begin{cases} \dot{z} &= Az + B\,\sigma(Cz) + Fu \\ q &= Hz \end{cases}$$

$$\mathcal{C}: \quad u = p + \epsilon - q$$

with master system $\mathcal{M}$, slave system $\mathcal{S}$, full static error feedback controller $\mathcal{C}$ and linear filter $\mathcal{R}$. For the master and slave systems, we consider the state vectors, system matrices and nonlinearity as described for the system (4.7). The message signal $r \in \mathbb{R}$, which is assumed to be binary valued is low pass filtered by $\mathcal{R}$. The master system is driven by the filtered message signal $d \in \mathbb{R}$ through the matrix $D \in \mathbb{R}^{n\times 1}$. The signal $p \in \mathbb{R}^l$ is transmitted through the channel, which is corrupted by the noise signal $\epsilon \in \mathbb{R}^l$. The low pass filter $\mathcal{R}$ has the state vector $m \in \mathbb{R}^{n_r}$ and system matrices $R \in \mathbb{R}^{n_r\times n_r}$, $S \in \mathbb{R}^{n_r\times 1}$, $T \in \mathbb{R}^{1\times n_r}$ and $U \in \mathbb{R}$. The slave system is controlled by the control signal $u \in \mathbb{R}^l$ through the matrix $F \in \mathbb{R}^{n\times l}$. The system is visualized in Figure 4.5. This basic scheme could be refined in terms of additional channel and noise

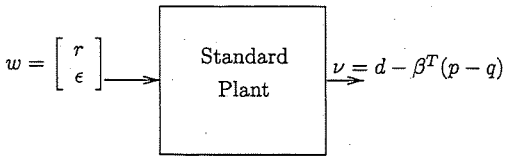


Figure 4.6: Control theoretic interpretation of the synchronization scheme by its standard plant representation, with exogenous input w and regulated output ν. The exogenous input consists of the filtered message signal d and the channel noise ϵ. The aim is to find a feedback matrix such that the influence of the exogenous input on the tracking error ν is minimized.

models. An error system in the sense of the autonomous case (4.8) can never be globally asymptotically stable for (4.63) because it is driven by the message r. Therefore another error signal concept has been proposed in [250, 243, 242] by defining as *tracking error* signal

$$\nu = d - \beta^T e \tag{4.64}$$

where $\beta = [1\ 0\ 0\ \cdots\ 0]^T$ selects the first component of $e = p - q$. The synchronization scheme is represented then in standard plant form (Figure 4.6) [165, 21]:

$$\mathcal{E}: \begin{cases} \dot{e} &= A_s e + HDTm + B_s\eta(C_s e; z) + HDUr - HF\epsilon \\ \dot{m} &= Rm + Sr \\ \nu &= -\beta^T e + Tm + Ur \end{cases} \tag{4.65}$$

with $A_s = HAH^{-1} - HF$, $B_s = HB$ and $C_s = CH^{-1}$.

In order to analyze the I/O properties of standard plant representation of the synchronization scheme, we consider the quadratic storage function:

$$\phi(\xi) = \xi^T P \xi, \ \ P = P^T > 0 \tag{4.66}$$

where $\xi = [e^T\ m^T]^T$ and a supply rate with finite L_2-gain γ

$$s(w,\nu) = \gamma^2 w^T w - \nu^T \nu \tag{4.67}$$

with regulated output ν and exogenous input w. As exogenous input we take $w = [r^T\ \epsilon^T]^T$, which consists of the reference input and the disturbance signal. The aim is here to design the feedback matrix F such that the tracking error signal (4.64) is minimized.

Theorem 4.8 (Nonlinear H_∞ synchronization using static controller) **[250]** *Suppose Assumption 4.1 holds and that there exists $\Lambda = \text{diag}\,\{\lambda_i\}$ a diagonal matrix with $\lambda_i \geq 0$ for $i = 1, 2, ..., n_h$ and $0 < P = P^T \in \mathbb{R}^{n_l+n_r \times n_l+l}$. Then a sufficient condition for dissipativity of the synchronization scheme (4.63) with respect to the quadratic storage function $\phi(\xi)$ (4.66) where $\xi = [e^T\ m^T]^T$ and supply rate (4.67) with finite L_2-gain γ, is given by matrix inequality*

$$Z = Z^T = \begin{bmatrix} Z_{11} & Z_{12} & Z_{13} & Z_{14} & Z_{15} \\ . & Z_{22} & Z_{23} & Z_{24} & Z_{25} \\ . & . & Z_{33} & 0 & 0 \\ . & . & . & Z_{44} & 0 \\ . & . & . & . & Z_{55} \end{bmatrix} < 0 \tag{4.68}$$

with

$$\begin{aligned} Z_{11} &= A_s P_{11} + P_{11} A_s + \beta\beta^T & Z_{12} &= A_s^T P_{12} + P_{11} HDT + P_{12} R - \beta T \\ Z_{13} &= P_{11} B_s + k C_s^T \Lambda & Z_{14} &= P_{11} HDU + P_{12} S - \beta U \\ Z_{15} &= -P_{11} HF & Z_{22} &= -P_{21} HDT + T^T D^T H^T P_{12} \\ Z_{23} &= P_{21} B_s & &\quad + P_{22} R + R^T P_{22} + T^T T \\ Z_{25} &= -P_{21} HF & Z_{24} &= P_{21} HDU + P_{22} S + T^T U \\ Z_{44} &= -\gamma^2 I + U^T U & Z_{33} &= -2\Lambda \\ Z_{55} &= -\gamma^2 I \end{aligned}$$

Proof: Dissipativity with finite L_2-gain γ is obtained if

$$\dot{\phi} \leq s(w,\nu), \ \forall w, \nu. \tag{4.69}$$

Taking the time derivative of the quadratic storage function, and appling the $\mathcal{S}$-procedure, using the inequalities of Eq. (4.10), we obtain

$$\begin{aligned} \dot{\phi} - s(w,\nu) &= \dot{\xi}^T \xi + \xi^T P \dot{\xi} - s(w,\nu) \\ &\leq (A_s e + B_s \eta + HDTm + HDUr - HF\epsilon)^T (P_{11} + P_{12} m) \\ &+ (Rm + Sr)^T (P_{21} e + P_{22} m) \\ &+ (e^T P_{11} + m^T P_{21})(A_* e + B_* \eta + H(DTm + DUr - F\epsilon)) \\ &+ (e^T P_{12} + m^T P_{22})(Rm + Sr) - \sum_i 2\lambda_i \eta_i (\eta_i - k c_i^T e) \\ &\quad - \gamma^2 (r^2 + \epsilon^t \epsilon) + (Tm - \beta^T e Ur)^T (Tm - \beta^T e + Ur) \\ &\leq \zeta^T Z \zeta < 0 \end{aligned} \tag{4.70}$$

where $\zeta = [e^T\ m^T\ \eta^T\ r^T\ \epsilon^T]^T$. The latter expression is negative for all nonzero ζ, if Z is a negative definite matrix. □

4.5.2 Dynamic output feedback

Consider now the master-slave synchronization scheme with dynamic output error feedback and identical master-slave Lur'e systems:

$$\mathcal{R}:\left\{\begin{array}{rcl}\dot{m} &=& Rm+Sr\\ d &=& Tm+Ur\end{array}\right.$$

$$\mathcal{M}:\left\{\begin{array}{rcl}\dot{x} &=& Ax+B\,\sigma(Cx)+Dd\\ p &=& Hx\end{array}\right. \tag{4.71}$$

$$\mathcal{S}:\left\{\begin{array}{rcl}\dot{z} &=& Az+B\,\sigma(Cz)+Fu\\ q &=& Hz\end{array}\right.$$

$$\mathcal{C}:\left\{\begin{array}{rcl}\dot{\rho} &=& E\rho+G(p+\epsilon-q)\\ u &=& M\rho+N(p+\epsilon-q)\end{array}\right.$$

with master system $\mathcal{M}$, slave system $\mathcal{S}$, linear dynamic output feedback controller $\mathcal{C}$ and linear filter $\mathcal{R}$. For the systems $\mathcal{M}$, $\mathcal{S}$ and $\mathcal{C}$ we consider the state vectors, system matrices and nonlinearity as described for the system (4.15). The signal $d\in\mathbb{R}$ is modulated through the matrix $D\in\mathbb{R}^n$. The low pass filter $\mathcal{R}$ is the same as for the scheme (4.63). For the synchronization scheme, we obtain the standard plant representation

$$\mathcal{E}:\left\{\begin{array}{rcl}\dot{e} &=& A_de-F_d\rho+D_dm+B\eta(Ce;z)+DU\tau-FN\epsilon\\ \dot{\rho} &=& G_de+E\rho+G\epsilon\\ \dot{m} &=& Rm+Sr\\ \nu &=& -\beta^Te+Tm+Ur\end{array}\right. \tag{4.72}$$

with the error signal $\dot{e}=x-z$, regulated output ν and $A_d=A-FNH$, $F_d=FM$, $D_d=DT$ and $G_d=GH$.

Theorem 4.9 (Nonlinear H_∞ synchronization using dynamic controller) [243] *Suppose Assumption 4.1 holds and that there exists* $\Lambda=\mathrm{diag}\,\{\lambda_i\}$ *a diagonal matrix with* $\lambda_i\geq 0$ *for* $i=1,2,...,n_h$ *and* $0<P=P^T\in\mathbb{R}^{(n+n_c+n_r)\times(n+n_c+n_r)}$. *Then a sufficient condition for dissipativity of the synchronization scheme (4.71) with respect to the quadratic storage function* $\phi(\xi)$ *(4.66) where* $\xi=[e^T\ \rho^T\ m^T]^T$ *and supply rate (4.67) with finite* L_2*-gain* γ *is*

given by matrix inequality

$$Z=Z^T=\begin{bmatrix}Z_{11}&Z_{12}&Z_{13}&Z_{14}&Z_{15}&Z_{16}\\ .&Z_{22}&Z_{23}&Z_{24}&Z_{25}&Z_{26}\\ .&.&Z_{33}&Z_{34}&Z_{25}&Z_{36}\\ .&.&.&Z_{44}&0&0\\ .&.&.&.&Z_{55}&0\\ .&.&.&.&.&Z_{66}\end{bmatrix}<0 \tag{4.73}$$

with

$$\begin{aligned}
Z_{11} &= A_dP_{11}+P_{11}A_d+G_d^TP_{21}\\
&\quad +P_{12}G_d+\beta\beta^T\\
Z_{13} &= A_d^TP_{13}+G_d^TP_{23}-P_{11}D_d\\
&\quad +P_{13}R-\beta T\\
Z_{15} &= P_{11}DU+P_{13}S-\beta U\\
Z_{22} &= E^TP_{22}+P_{22}E-F_d^TP_{12}-P_{21}F_d\\
Z_{24} &= P_{21}B\\
Z_{26} &= P_{22}G-P_{21}FN\\
Z_{34} &= P_{31}B\\
Z_{36} &= P_{32}G-P_{31}FN\\
Z_{55} &= -\gamma^2I+U^TU\\
Z_{12} &= A_d^TP_{12}+G_d^TP_{22}-P_{11}F\\
&\quad +P_{12}E\\
Z_{14} &= P_{11}B+kC^T\Lambda\\
Z_{16} &= P_{12}G-P_{11}FN\\
Z_{23} &= E^TP_{23}-F_dP_{13}+P_{21}D_d\\
&\quad +P_{23}R\\
Z_{25} &= P_{21}DU+P_{23}S\\
Z_{33} &= P^TP_{33}+P_{33}R+D_d^TP_{13}\\
&\quad +P_{31}D_d+T^TT\\
Z_{44} &= -2\Lambda\\
Z_{66} &= -\alpha^2\gamma^2I.
\end{aligned}$$

Proof: By taking the quadratic storage function and supply rate with finite L_2 gain, it is straightforward to find the above condition using a similar procedure as in the previous proof and imposing $\dot{V}\leq[e^T\ \rho^T\ m^T\ \eta^T\ r^T\ \epsilon^T]\ Z\ [e^T\ \rho^T\ m^T\ \eta^T\ r^T\ \epsilon^T]^T$ □

4.6 Robust nonlinear H_∞ synchronization

In this section, the influence of parameter mismatch between the master and slave systems is studied with respect to the method of nonlinear H_∞ synchronization. Robust synchronization has already been presented for the master-slave system of Section 4.3, which can be considered as a step toward robust synchronization. Robust H_∞ synchronization has been considered for the case of full static state feedback and linear dynamic output error feedback by Suykens *et al.* [242].

4.6.1 Full static state error feedback

Consider the nonlinear H_∞ synchronization scheme with full static state error feedback for non-identical master-slave systems:

$$\mathcal{R}: \left\{ \begin{array}{rcl} \dot{m} & = & Rm + Sr \\ d & = & Tm + Ur \end{array} \right.$$

$$\mathcal{M}: \left\{ \begin{array}{rcl} \dot{x} & = & A_1 x + B_1\, \sigma(C_1 x) + Dd \\ p & = & Hx \end{array} \right. \tag{4.74}$$

$$\mathcal{S}: \left\{ \begin{array}{rcl} \dot{z} & = & A_2 z + B_2\, \sigma(C_2 z) + Fu \\ q & = & Hz \end{array} \right.$$

$$\mathcal{C}: \quad u = p + \epsilon - q$$

with master system $\mathcal{M}$, slave system $\mathcal{S}$, full static error feedback controller $\mathcal{C}$ and linear filter $\mathcal{R}$. The linear filter and controller are as described in the scheme (4.63). The master and slave system are non-identical Lur'e systems which are described in the scheme (4.21). The master system is driven by the filtered message signal $d \in \mathbb{R}$ through the matrix $D \in \mathbb{R}^{n\times 1}$. The signal $p \in \mathbb{R}^n$ is transmitted through the channel, which is corrupted by the noise signal $\epsilon \in \mathbb{R}^n$. As in the previous Section 4.5.1, we define the tracking error (4.64) where $\beta = [1\ 0\ 0\ \cdots\ 0]^T$ selects the first component of $e = p - q$. One obtains the standard plant representation

$$\mathcal{E}: \left\{ \begin{array}{ll} \dot{e} & = A_s e + B_s \eta(C_s e; z) + H(DTm + DUr - F\epsilon + \varphi(x)) \\ \dot{m} & = Rm + Sr \\ \nu & = -\beta^T e + Tm + Ur \end{array} \right. \tag{4.75}$$

with $A_s = HA_2H^{-1} - HF$, $B_s = HB_2$, $C_s = C_2H^{-1}$ and $\varphi(x)$ as defined in Eq. (4.24). In the standard plant representation (4.75) the influence of the parameter mismatch is collected in the term $\varphi(x)$ (4.24) like in Section 4.3.1. Here $\varphi(x)$ is thought as an input to the standard plant (4.75).

Because of the external input r, the Assumption 4.2 needs to be reassessed.

Assumption 4.3 *[242] The master system $\mathcal{M}$ is input to state stable in the sense that there exist initial states x_0 and a positive real constant δ such that*

$$\|x(t)\|_2 \leq \delta, \quad \forall t \in [0, \infty), \tag{4.76}$$

for all continuous time reference inputs r which satisfy $\|r\|_\infty = \sup_{t\geq 0} |r(t)| \leq 1$.

Using the expression for $\varphi(x)$ one obtains

$$\|\varphi(x)\|_2 \leq \beta_m \|x\|_2, \quad \forall x \tag{4.77}$$

with $\beta_m = \|\Delta A\|_2 + k\|B_1\|_2\|C_1\|_2 + k\|B_2\|_2\|C_2\|_2$ and $\Delta A = A_1 - A_2$. From Assumption 4.3, we have $\|\varphi(x)\|_2 \leq \beta_m\, \delta,\ \forall x$. For the upper bound, we will consider a positive constant scaling factor α:

$$\|\varphi(x)\|_2 \leq \alpha \beta_m \delta. \tag{4.78}$$

In order to analyze the I/O properties of the standard plant (4.75), we consider the quadratic storage function (4.66) where $\xi = [e^T\ m^T]^T$ and a supply rate with finite L_2-gain γ (4.67) with regulated output ν and exogenous input $w = [r^T\ \epsilon^T\ \beta_m\delta]^T$. The exogenous input consists of the reference input, disturbance signals (such as in Section 4.3) and constant signal related to the parameter mismatch between master and slave systems.

Theorem 4.10 (Robust nonlinear $\mathbf{H_\infty}$ synchronization using a static controller) [242] *Suppose Assumption 4.1 and 4.3 hold and that there exists $\Lambda = \mathrm{diag}\{\lambda_i\}$ a diagonal matrix with $\lambda_i \geq 0$ for $i = 1, 2, ..., n_h$ and $0 < P = P^T \in \mathbb{R}^{n+n_r\times n+n_r}$. Then a sufficient condition for dissipativity of the synchronization scheme (4.74) with respect to the quadratic storage function $\phi(\xi)$ (4.66) where $\xi = [e^T\ m^T]^T$ and supply rate (4.67) with finite L_2-gain γ is given by matrix inequality*

$$Y = Y^T = \left[\begin{array}{ccc|c} & & & P_{11}H \\ & & & P_{21}H \\ & Z & & 0 \\ & & & 0 \\ & & & 0 \\ \hline . & . & . & -\gamma I/\alpha^2 \end{array}\right] < 0 \tag{4.79}$$

where Z is given by Eq. (4.68).

Proof: One checks the condition $\dot{\phi} - s(w, \nu) \leq 0$ and applies the $\mathcal{S}$-procedure using the sector condition on η (Eq. (4.10)) together with the condition (4.78) which is expressed as $\beta_m^2\delta^2 - \varphi^T\varphi/\alpha^2 \geq 0$. Then one obtains

$$\dot{\phi} - s(w, \nu) + 2\eta^T\Lambda(\eta - kC_2e) + \tau(\beta_m^2\delta^2 - \varphi^T\varphi/\alpha^2) < 0. \tag{4.80}$$

Using the expression for the supply rate $s(w,\nu) = \gamma^2(r^T r + \epsilon^T\epsilon + \beta_m^2\delta^2) - \nu^T\nu$ and choosing $\tau = \gamma^2$ this condition becomes

$$\dot{\xi}^T P\xi + \xi^T P\dot{\xi} - \gamma^2(r^T r + \epsilon^T\epsilon) + \nu^T\nu + 2\eta^T\Lambda(\eta - kC_2 e) - \gamma^2\varphi^T\varphi/\alpha^2 < 0 \quad (4.81)$$

The latter expression can be written as the quadratic form $\zeta^T Y\zeta < 0$ with $\zeta = [e^T\ m^T\ \eta^T\ r^T\ \epsilon^T\ \varphi^T]^T$. This expression is negative for all ζ if Y is negative definite. □

4.6.2 Dynamic output feedback

Consider the nonlinear H_∞ synchronization scheme with full static state error feedback for non-identical master-slave systems:

$$\mathcal{R}:\begin{cases} \dot{m} &= Rm + Sr \\ d &= Tm + Ur \end{cases}$$

$$\mathcal{M}:\begin{cases} \dot{x} &= A_1 x + B_1\,\sigma(C_1 x) + Dd \\ p &= Hx \end{cases} \quad (4.82)$$

$$\mathcal{S}:\begin{cases} \dot{z} &= A_2 z + B_2\,\sigma(C_2 z) + Fu \\ q &= Hz \end{cases}$$

$$\mathcal{C}:\begin{cases} \dot{\rho} &= E\rho + G(p + \epsilon - q) \\ u &= M\rho + N(p + \epsilon - q) \end{cases}$$

with master system $\mathcal{M}$, slave system $\mathcal{S}$, linear dynamic output feedback controller $\mathcal{C}$ and linear filter $\mathcal{R}$. The linear filter and controller are as described in the scheme (4.71). The master and slave system are two non-identical Lur'e systems which are described by the scheme (4.21). The master system is driven by the filtered message signal $d \in \mathbb{R}$ through the matrix $D \in \mathbb{R}^n$. The signal $p \in \mathbb{R}^l$ is transmitted through the channel, which is corrupted by the noise signal $\epsilon \in \mathbb{R}^l$.

Defining the same tracking error as in the previous section, the following standard plant representation is obtained

$$\mathcal{E}:\begin{cases} \dot{e} &= A_d e + B\eta(Ce; z) + D_d m - F_d\rho + DUr - FN\epsilon + \varphi(x) \\ \dot{\rho} &= G_d e + E\rho + G\epsilon \\ \dot{m} &= Rm + Sr \\ \nu &= -\beta^T e + Tm + Ur \end{cases} \quad (4.83)$$

where $e = x - z$ and $A_d = A - FNH$, $F_d = FM$, $D_d = DT$ and $G_d = GH$. This new standard plant representation includes the standard plant representation (4.72) and an additional term $\varphi(x)$ which contains the influence of the parameter mismatch. This additional term will be thought as an input to the standard plant, like in the previous section.

The I/O properties of the standard plant representation are analyzed using the quadratic storage function (4.66) where $\xi = [e^T\ \rho^T\ \eta^T]^T$, the supply rate with finite L_2-gain (4.67) and exogenous input $w = [r^T\ \epsilon^T\ \beta_m\delta]^T$.

Theorem 4.11 (Robust nonlinear H_∞ synchronization using dynamic controller) [242] *Suppose Assumption 4.1 and 4.3 hold and that there exists $\Lambda = \mathrm{diag}\{\lambda_i\}$ a diagonal matrix with $\lambda_i \geq 0$ for $i = 1, 2, ..., n_h$ and $0 < P = P^T \in \mathbb{R}^{n+n_r+n_c \times n+n_r+n_c}$. Then a sufficient condition for dissipativity of the synchronization scheme (4.82) with respect to the quadratic storage function $\phi(\xi)$ (4.66) where $\xi = [e^T\ \rho^T\ m^T]^T$ and supply rate (4.67) with finite L_2-gain γ is given by matrix inequality*

$$Y = Y^T = \left[\begin{array}{c|c} & P_{11} \\ & P_{21} \\ Z & P_{31} \\ & 0 \\ & 0 \\ & 0 \\ \hline \cdots\cdots & -\gamma I/\alpha^2 \end{array}\right] < 0 \quad (4.84)$$

where Z is given in Eq. (4.73).

Proof: It is straightforward to find the above condition using a similar procedure as in the previous proof and imposing $\dot{V} \leq [e^T\ \rho^T\ m^T\ \eta^T\ r^T\ \epsilon^T]\ Y\ [e^T\ \rho^T\ m^T\ \eta^T\ r^T\ \epsilon^T]^T$. □

4.7 Impulsive synchronization

In [315] a theory of impulsive differential equations [156] has been applied to chaos control and synchronization of Chua's circuit. For Lur'e systems certain extensions have been presented in [255, 254, 256]. In impulsive control the control signal is applied at discrete time instants, which are allowed to be non-equidistant in time. The method of impulsive synchronization has been applied to chaotic digital code-division multiple access (CDMA) systems [316].

4.7.1 State feedback case

Here we illustrate the main ideas by hand of the synchronization scheme

$$\left\{\begin{array}{lrcll} \mathcal{M}: & \dot{x} & = & Ax + B\sigma(Cx) & \\ \mathcal{S}: & \dot{z} & = & Az + B\sigma(Cz), & t \neq \tau_i \\ \mathcal{C}: & \Delta z & = & K(x-z), & t = \tau_i. \end{array}\right. \tag{4.85}$$

For the impulsive control law $\mathcal{C}$, a set of discrete time instants τ_i are considered where $0 < \tau_1 < \tau_2 < ... < \tau_i < \tau_{i+1} < ...$ At the time instants τ_i, instantaneous jumps in the state variable z are imposed: $\Delta z|_{t=\tau_i} = z(\tau_i^+) - z(\tau_i^-)$. One has the error system

$$\left\{\begin{array}{lcll} \dot{e} & = & Ae + B\eta(Ce;z), & t \neq \tau_i \\ \Delta e & = & -K(x-z), & t = \tau_i \end{array}\right. \tag{4.86}$$

with $\Delta e = \Delta x - \Delta z$ with $\Delta x = 0$ for the master system.

Theorem 4.12 (Impulsive synchronization: state feedback) [255] *Let $\Lambda = \text{diag}\ \{\lambda_i\}$ be a diagonal matrix with $\lambda_i \geq 0$ for $i = 1, 2, ..., n_h$, $0 < P = P^T \in \mathbb{R}^{n \times n}$ and α, β scalars. Then a sufficient condition for impulsive synchronization of the master slave system (4.85) is given by following set of inequalities*

$$\left[\begin{array}{cc} A^TP + PA - \alpha P & PB + C^T\Lambda \\ B^TP + \Lambda C & -2\Lambda \end{array}\right] \leq 0$$

$$(I-K)^TP(I-K) < \beta P$$

$$(I-K)^T(I-K) < I$$

$$\alpha(\tau_{i+1} - \tau_i) + \log\beta < 0.$$

Proof: By employing the Lyapunov function $V(e)$ (4.5). According to the theory of impulsive differential equations as discussed in [156, 315] it is sufficient then to prove that

$$\left\{\begin{array}{lll} \dot{V} \leq \alpha V, & \alpha > 0, & t \neq \tau_i \\ V(\zeta + \Delta\zeta) < \beta V, & \beta > 0, & t = \tau_i \\ \|\zeta + \Delta\zeta\|_2 < \|\zeta\|_2, & & t = \tau_i \\ \alpha(\tau_{i+1} - \tau_i) + \log\beta < 0. & & \end{array}\right. \tag{4.87}$$

After exploiting the sector condition on η the set of inequalities is derived as a sufficient condition for synchronization. □

4.7.2 Measurement feedback case

We consider the following master-slave synchronization scheme

$$\mathcal{M}: \left\{\begin{array}{lcl} \dot{x} & = & Ax + B\sigma(Cx) \\ p & = & Lx \end{array}\right.$$

$$\mathcal{S}: \left\{\begin{array}{lcl} \dot{z} & = & Az + B\sigma(Cz) \\ q & = & Lz \end{array}\right.$$

$$\mathcal{C}: \left\{\begin{array}{lcll} \dot{\xi} & = & E\xi + F(p-q) & \\ & & +W_F\sigma(V_{F_1}\xi + V_{F_2}(p-q)) & \tau \neq \tau_k \\ \Delta z & = & D_1u, & \tau = \tau_k \\ \Delta\xi & = & D_2v, & \tau \neq \tau_k \\ u & = & G_1\xi + H_1(p-q) & \\ v & = & G_2\xi + H_2(p-q) & \end{array}\right. \tag{4.88}$$

which consists of a master $\mathcal{M}$, a slave system $\mathcal{S}$ and a controller $\mathcal{C}$. $\mathcal{M}$ and $\mathcal{S}$ are identical Lur'e systems. The output or measurement vectors of $\mathcal{M}$ and $\mathcal{S}$ are $p, q \in \mathbb{R}^l$ with $l \leq n$ and $L \in \mathbb{R}^{l \times n}$. For the sake of generality, a nonlinear dynamic output feedback controller of the Lur'e form is taken for the state equation with state vector $\xi \in \mathbb{R}^{n_\xi}$. By means of the matrices D_1 and D_2 it is decided on which state equations the impulsive controls $u \in \mathbb{R}^{m_z}$ and $v \in \mathbb{R}^{m_\xi}$ are applied. The output difference $p - q$ is taken as input of the controller $\mathcal{C}$. The matrices of the controller are of dimension $E \in \mathbb{R}^{n_\xi \times n_\xi}$, $F \in \mathbb{R}^{n_\xi \times l}$, $W_F \in \mathbb{R}^{n_\xi \times n_{h_\xi}}$, $V_{F_1} \in \mathbb{R}^{n_{h_\xi} \times n_\xi}$, $V_{F_2} \in \mathbb{R}^{n_{h_\xi} \times l}$, $D_1 \in \mathbb{R}^{n_z \times m_z}$, $D_2 \in \mathbb{R}^{n_\xi \times m_\xi}$, $G_1 \in \mathbb{R}^{m_z \times n_\xi}$, $G_2 \in \mathbb{R}^{m_\xi \times n_\xi}$, $H_1 \in \mathbb{R}^{m_z \times l}$, $H_2 \in \mathbb{R}^{m_\xi \times l}$ where n_{h_ξ} is the number of hidden units in the Lur'e system of $\mathcal{C}$. The output error system for e_L becomes

$$\mathcal{E}: \left\{\begin{array}{lcll} \dot{e_L} & = & LAe + LB\eta(Ce;z), & \tau \neq \tau_k \\ \dot{\xi} & = & E\xi + F(p-q) & \\ & & +W_F\sigma(V_{F_1}\xi + V_{F_2}(p-q)), & \tau \neq \tau_k \\ \Delta e_L & = & -LD_1u, & \tau = \tau_k \\ \Delta\xi & = & D_2v, & \tau = \tau_k \\ u & = & G_1\xi + H_1(p-q) & \\ v & = & G_2\xi + H_2(p-q) & \end{array}\right. \tag{4.89}$$

where the synchronization error is defined as $e = x - z$ for the state vectors and $e_L = p - q = Le$ for the outputs and also $\Delta e_L = -L\Delta z$.

Theorem 4.13 (Impulsive synchronization: measurement feedback) [254, 256] *Let Λ and Γ be diagonal matrices with positive diagonal elements,*

$\alpha > 0$ and β are scalars and $P = Q^T Q > 0$. Then a sufficient condition for impulsive synchronization of the master slave system (4.89) with measurement feedback is given by following set of inequalities

$$\begin{aligned} Z &\leq 0 \\ M^T P M &< \beta P \\ M^T M &< I \\ \alpha(\tau_{k+1} - \tau_k) + \log\beta &< 0 \end{aligned}$$

where $Z = Z^T$ and matrix blocks

$$\begin{aligned} Z_{11} =& AP_{11} + L^T P_{11} A + L^T P_{21} FL + L^T F^T P_{21} L \\ & - \alpha_* L^T P_{11} L \\ Z_{12} =& A^T P_{12} + L^T P_{12} E + L^T F^T P_{22} - \alpha_* L^T P_{12} \\ Z_{13} =& L^T P_{11} LB + C^T \Lambda \\ Z_{14} =& L^T P_{12} W_F + L^T V_{F_2}^T \Gamma \\ Z_{33} =& -2\Lambda \\ Z_{44} =& -2\Gamma \\ Z_{23} =& P_{21} LB \\ Z_{24} =& P_{22} W_F + V_{F_1}^\Gamma \\ Z_{34} =& 0 \end{aligned}$$

and $M = \begin{bmatrix} I - LD_1 H_1 & -LD_1 G_1 \\ D_2 H_2 & I + D_2 G_2 \end{bmatrix}$.

Proof: In order to derive a sufficient condition for global asymptotic stability of the error system $\mathcal{E}$, the Lyapunov function

$$V(e_L, \xi) = [e_L^T \; \xi^T] \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} e_L \\ \xi \end{bmatrix} \tag{4.90}$$

is taken. According to [156, 315] it is sufficient then to prove (4.87). After exploiting the sector condition on η the first condition which is $Z \leq 0$, can be formulated as given in the Lemma. In the order to express the other conditions in (4.87) as a matrix inequality, one writes

$$\zeta + \Delta\zeta = \begin{bmatrix} e_L \\ \xi \end{bmatrix} + \begin{bmatrix} \Delta e_L \\ \Delta\xi \end{bmatrix} = M \begin{bmatrix} e_L \\ \xi \end{bmatrix}. \tag{4.91}$$

This yields the matrix inequality $M^T P M < \beta P$ and $M^T M < I$. □

4.8 Controller design

In this Section, we explain how to design the controllers (C) for achieving synchronization. All the given synchronization criteria in this Chapter have

been expressed in terms of matrix inequalities which lead to optimization based design [82, 190, 21]. Therefore, the presented synchronization criteria can be related to optimization problems. The optimization problem aims at finding a controller which minimizes the objective function such that it also satisfies some constraint function. In general, the optimization problems which are presented in this section, are non-convex. Non-differentiability might occur when the two largest eigenvalues of Y or Z coincide. Convergent algorithms for this kind of non-differentiable optimization problem have been described in [200].

4.8.1 Master-slave synchronization

The static controller of the master-slave synchronization scheme (4.7) can be designed on the basis of matrix inequality Eq.(4.11) by solving the following nonlinear optimization problem [253]:

$$\min_{F,P,\Lambda} \lambda_{\max}(Z(F,P,\Lambda)) \quad \text{such that} \quad \begin{cases} P & = P^T > 0 \\ \Lambda & \geq 0. \end{cases} \tag{4.92}$$

In order to bound the norm of the feedback matrix, the constraint $\|F\|_2 < c$ can be included as constraint in the optimization problem. The constraint $P > 0$ can be eliminated by considering the parametrization $P = Q^T Q$. A similar idea applies to Λ. Then the optimization problem becomes:

$$\min_{F,Q,\Lambda} \lambda_{\max}(Z(F,Q,\Lambda)) \quad \text{such that} \quad \|F\|_2 < c. \tag{4.93}$$

When the maximal eigenvalue of Z is negative, a feasible for F, Q, Λ point to the matrix inequality is obtained, resulting into a controller (in this case, it is a feedback matrix F) which synchronizes the master-slave system (4.7).

The dynamic output feedback controller for the scheme (4.15) can be designed on the basis of matrix inequality Eq. (4.18) by solving the following nonlinear optimization problem [253]:

$$\min_{E,G,M,N,P,\Lambda} \lambda_{\max}(Y(E,G,M,N,P,\Lambda)) \quad \text{such that} \quad \begin{cases} P & = P^T > 0 \\ \Lambda & \geq 0. \end{cases} \tag{4.94}$$

This problem can be parameterized in a similar way. Hence, we have the following optimization problem:

$$\min_{E,G,M,N,Q,\Lambda} \lambda_{\max}(Y(E,G,M,N,Q,\Lambda)). \tag{4.95}$$

The resulting E, G, M, N matrices resulting from the optimization problem (4.95) given a dynamic output feedback controller for the scheme (4.15) that synchronizes the master-slave system.

4.8.2 Robust synchronization

The following uncertainty relation between the synchronization error bound ϵ and the parameter mismatch of the master and slave systems was derived by Suykens *et al.*: [241]

$$\frac{\epsilon}{\beta} = c_0 \tag{4.96}$$

where the positive real constant c_0 is equal to $2\delta\sigma_{max}(P)\kappa(P)/\alpha$. The robustness criterion is based on this uncertainty relation. Therefore, the design of the controller aims at minimizing the constant c_0 in the uncertainty relation such that the matrix inequality (4.28) is satisfied. This corresponds to the following nonlinear optimization problem [241]:

$$\min_{F,P,\Lambda,\alpha} \frac{\sigma_{max}(P)\kappa(P)}{\alpha} \quad \text{such that} \quad \begin{cases} Y(F,P,\Lambda,\alpha) < 0 \\ P = P^T > 0 \\ \Lambda \geq 0. \end{cases} \tag{4.97}$$

For the dynamic feedback case (Section 4.3.2), the volume of the ball $\mathcal{B}_1$ is minimized [241]. With respect to the ball $\mathcal{B}_1$ the uncertainty relation (4.96) has the positive real constant c_0 is equal to $\eta_1 = \frac{\tau_1}{2\alpha_1} + \sqrt{\frac{\tau_1^2}{\alpha_1^2} + \frac{\tau_2^2}{\alpha_1\alpha_2}}$. The design for the dynamic feedback case can be done then as

$$\min_{E,G,M,N,P,\Lambda,\alpha_1,\alpha_2} \frac{\sigma_{max}(P_{11})}{\alpha_1} + \frac{\sigma_{max}(P_{12})}{\alpha_2}$$
$$\text{such that} \quad \begin{cases} Z(E,G,M,N,P,\Lambda,\alpha_1,\alpha_2) < 0 \\ \kappa(P) \leq \kappa_0 \\ P = P^T > 0 \\ \Lambda \geq 0 \end{cases} \tag{4.98}$$

where κ_0 is a user-defined upper bound [241].

4.8.3 Synchronization with time-delay

The static controller of the master-slave synchronization scheme with time delay can be designed on the matrix inequality (4.45) by solving the following nonlinear optimization problem [305]:

$$\min_{F,P,Q,\Lambda} \lambda_{\max}(Y(F,P,Q,\Lambda)) \quad \text{such that} \quad \begin{cases} P = P^T > 0 \\ Q = Q^T > 0 \\ \Lambda \geq 0. \end{cases} \tag{4.99}$$

When the maximal eigenvalue of Y is negative, a feasible point to the matrix inequality (4.45) is obtained, resulting into a controller that synchronizes the master-slave system (4.42) whatever the time-delay τ.

For delay-dependent synchronization, the controller can be designed based on the matrix inequality (4.62) by solving the following nonlinear optimization problem [305]:

$$\min_{F,P,Q,\Lambda,r_1,r_2,r_3} \lambda_{\max}(Y(F,P,Q,\Lambda,r_1,r_2,r_3))$$
$$\text{such that} \quad \begin{cases} P = P^T > 0 \\ Q = Q^T > 0 \\ \Lambda \geq 0 \\ r_1, r_2, r_3 > 0. \end{cases} \tag{4.100}$$

4.8.4 Nonlinear H_∞ synchronization

A controller for a given nonlinear plant is designed by considering a supply rate with finite L_2-gain in nonlinear H_∞ control theory [270]. The optimal H_∞ control law corresponds then to the minimal L_2-gain γ that is achievable for the system such that the closed loop system is dissipative. The optimal solution is characterized here by matrix inequalities, as is shown in Theorems 4.8 and 4.9. The nonlinear H_∞ synchronization problem seeks to find the linear static controller which minimizes the L_2-gain γ with respect to the matrix inequalities (4.68) [250]:

$$\min_{F,P,\Lambda,\gamma} \gamma \quad \text{such that} \quad \begin{cases} Z(F,P,\Lambda,\gamma) < 0 \\ P = P^T > 0 \\ \Lambda \geq 0. \end{cases} \tag{4.101}$$

Similar to static controller, the linear dynamic output feedback controller is designed by following nonlinear optimization problem [243]:

$$\min_{E,G,M,N,P,\Lambda,\gamma} \gamma \quad \text{such that} \quad \begin{cases} Z(E,G,M,N,P,\Lambda,\gamma) < 0 \\ P = P^T > 0 \\ \Lambda \geq 0. \end{cases} \tag{4.102}$$

4.8.5 Robust nonlinear H_∞ synchronization

In the robust nonlinear H_∞ synchronization, the scaling factor α is maximized, in order to achieve maximal robustness with respect to parameter mismatch between the master-slave systems, as follows from (4.78) [242]. Using a penalty method [82] the robust nonlinear H_∞ synchronization problem for static controller can be formulated as follows [242]:

$$\min_{F,P,\Lambda,\gamma,\alpha} \gamma + c\frac{1}{\alpha} \quad \text{such that} \quad \begin{cases} Y(F,P,\Lambda,\gamma,\alpha) < 0 \\ P = P^T > 0 \\ \Lambda \geq 0. \end{cases} \tag{4.103}$$

where c is a positive real constant. Similar to a static controller, the linear dynamic output feedback controller is designed by following nonlinear optimization problem [242]:

$$\min_{E,G,M,N,P,\Lambda,\gamma,\alpha} \gamma + c\frac{1}{\alpha}$$
$$\text{such that} \quad \begin{cases} Y(E,G,M,N,P,\Lambda,\gamma,\alpha) < 0 \\ P = P^T > 0 \\ \Lambda \geq 0. \end{cases} \tag{4.104}$$

4.9 Examples

Many of the schemes discussed in this Chapter have been verified on many examples both in simulation and with real electronic circuit implementations, such as for Chua's circuits, n-scroll attractors from generalized Chua's circuit and hyperchaotic systems that consist of two unidirectionally coupled Chua's circuits. These circuits and their implementations have been already discussed in Chapter 3.

4.9.1 Master-slave synchronization of two unidirectionally coupled Chua's circuits

Master-slave synchronization using full static state error feedback and dynamic output feedback cases have been experimentally confirmed by observations done on computer simulations for Chua's circuit in [247] and [240], respectively. Instead of repeating the same experiment on a real electronic circuit implementation, we consider to apply the master-slave synchronization method into two nonlinearly coupled Chua's circuits (see Figure 4.7) which has been studied in [297, 226, 227].

This experiment can also be related to the study in Section 3.4.2. So we have two unidirectionally coupled CNN cells made of Chua's circuits as discussed in Section 2.2.5.3. Therefore, this experiment is an exercise on master-slave synchronization for Section 4.2.1 and n-scroll hypercubes attractors from Section 3.4.2. The cell dynamics of two CNN cells consisting of Chua's circuits ($\mathcal{M}$ and $\mathcal{S}$) and the dynamics of the master-slave system are described by

$$\begin{aligned} \mathcal{M}: \quad \dot{x} &= Ax + B\,\sigma(Cx) \\ \mathcal{S}: \quad \dot{z} &= Az + B\,\sigma(Cz) + u \\ \mathcal{C}: \quad u &= F_\sigma(\sigma(Cx) - \sigma(Cz)) + FH(x - z) \end{aligned} \tag{4.105}$$

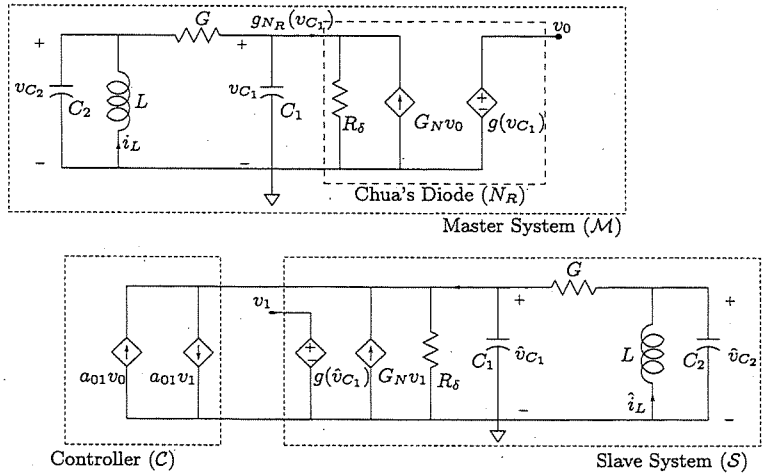


Figure 4.7: Two nonlinearly coupled Chua's circuits (3.29) with nonlinearity (3.30) ($g(v_{C_1})$). It might also be thought as two nonlinearly coupled generalized Chua's circuits with nonlinearity (3.43).

with state vector $x = [v_{C_1}\ v_{C_2}\ i_L]^T$ and $z = [\hat{v}_{C_1}\ \hat{v}_{C_2}\ \hat{i}_L]^T$. The system matrices of the Lur'e systems $(\mathcal{M}, \mathcal{S})$ and the nonlinearity $\sigma(\cdot)$ are given in Eq. (3.89). Controller $\mathcal{C}$ equals

$$F_\sigma = a_{01}B,\ F = a_{01}\alpha\overline{m}_1,\ H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{4.106}$$

where $a_{01} \in \mathbb{R}$. We can consider the above representation as a master-slave system and use the synchronization criterion (4.11). The parameter value a_{01} has to be considered as a design parameter for the synchronization method. Like in Section 4.2.1, investigating synchronization of the scheme is done by stability analysis of the error system

$$\mathcal{E}:\ \dot{e} = (A - FH)e + (B - F_\sigma)\,\eta(Ce; z) \tag{4.107}$$

where $\eta(Ce; z) = \sigma(Ce + Cz) - \sigma(Cz)$. The nonlinearity η belongs to sector [0, 1] [286, 63]. Hence Assumption 4.1 is satisfied by the nonlinearity η. This error system (4.107) is the same as the error system (4.8) when $B - F_\sigma$ is taken as B. Therefore, we can use Theorem 4.1. In order to find a_{01} together with the matrices P and Λ such that Theorem 4.1 holds, the constrained nonlinear optimization problem (4.92) has been solved. In Matlab, this optimization problem was solved by means of sequential quadratic programming for Chua circuit (3.3) with the parameters $\{18, 28.56, -8/7, -5/7\}$ and a design parameter value (a_{01}) has been obtained. Alternatively, one may prefer to solve convex subproblems for the LMIs. As a starting point for iterative procedure, a random controller parameter has been chosen. For the matrix Q, a square random matrix was chosen. The matrix Λ has been initialized as $100I$. As a solution to the optimization problem (4.93) we obtain $a_{01} = 0.9$. In the realization, the synthesis of dependent sources based on the voltage controlled voltage sources approach have been used for realization of two identical Chua's circuits. This approach is already given in Section 3.1.2 and a double scroll attractor has been observed on the circuit. Figure 4.9(a) shows a (v_{C_1}, v_{C_2}) plot which is phase portrait of the master system in Eq. (4.105). These two Chua's circuits are coupled with voltage controlled current source (VCCS) which is shown in Figure 4.7 with ideal circuit elements. The electronic realization of VCCS is given in Figure 4.8. Currents and time were scaled by a factor 10000. In order to obtain $a_{01} = 0.9$, R_{01} has to be chosen $100/9k\Omega$.

Figure 4.9(b) shows a $(v_{C_1}, \hat{v}_{C_1})$ plot, which corresponds to the first state variables for master and slave systems (4.105), when the master and slave systems are not connected ($u = 0$). The result of master-slave synchronization is shown in the $(v_{C_1}, \hat{v}_{C_1})$-plot in Figure 4.9(c) for the obtained a_{01}.

By means of Chua's circuit, Suykens *et al.* [241] illustrated that a relatively

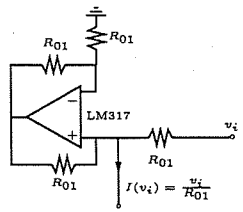


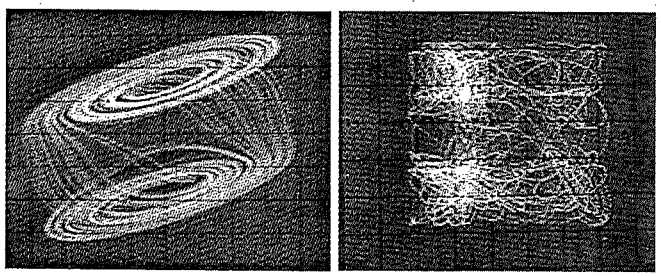
Figure 4.8: An electronic realization of a voltage controlled current source ($a_{01} = 1/R_{01}$).

large parameter mismatch between Lur'e systems can be allowed in order to preserve synchronization with a relatively small synchronization error bound. Both the static state error feedback and dynamic output error feedback have been studied for robust synchronization in [241].

4.9.2 Experimental confirmation of time-delay synchronization scheme

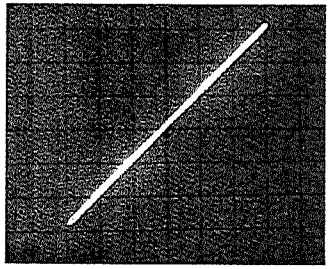
For schemes with time delay (4.42), Chua's circuit and hyperchaotic attractors have been tested for delay-independent synchronization which can be formulated as Eq. (4.99) such that Theorem 4.5 holds. It turnouts that Theorem 4.5 is useless for these examples. Instead, delay-dependent master-slave synchronization criteria need to be applied and controllers are designed by solving the constrained nonlinear optimization problem (4.100). Sequential quadratic programming has been applied in Matlab's optimization toolbox for different values of τ.

For Chua's circuit with the parameters $\{9, 14.28, -8/7, -5/7\}$ which can be represented in Lur'e form (3.85) by Eq. (3.89), the matrix $G = [6.0229\ \ 1.3367\ \ -2.1264]^T$ stabilizes the error system for $\tau \in [0, 0.039]$. In the experiments $H = [1\,0\,0]$ was chosen which means that the master system is connected to the slave system with the first state variable only. The synchronization scheme (4.42) has been modelled in Matlab Simulink and following simulation results have been obtained. The first observation is given in Figures 4.10(a)-4.10(c) for maximum delay $\tau^* = 0.039$. A second observation is given in Figures 4.10(e)-4.10(f) for a smaller delay $\tau = 0.01$ than the maximum delay. During the simulations it has been observed that two Chua's circuits synchronize until a maximal value of $\tau = 0.21$ for the same feedback matrix. Synchronization could not be observed for τ values larger than 0.21. The simulation result for $\tau = 0.22$ is given in Figures 4.10(g)-4.10(i). As initial conditions of the master and slave



(a)

(b)



(c)

Figure 4.9: Scope pictures for master-slave synchronization using Chua's circuit implementations: (a) a double scroll attractor observed from the master system. The y-axis corresponds to $v_{C_1}(t)$ while the x-axis corresponds to $v_{C_2}(t)$ $(0.5V/div - 0.2V/div)$. The plot of $v_{C_1}(t)$ versus $\hat{v}_{C_1}(t)$: (b) The master and slave systems are uncoupled $(0.5V/div - 0.5V/div)$ and (c) the result of the master-slave synchronization for the obtained controller $(0.5V/div - 0.5V/div)$.

systems were taken $x(0) = [-0.2 \ -0.33 \ 0.2]^T$, $z(0) = [0.5 \ -0.1 \ 0.66]^T$ in the simulations. As a hyperchaotic attractor, we consider a 2-scroll hypercube attractor (3.80) which consists of two unidirectionally coupled ($k = 0.01$) Chua circuits with the parameters $\{9, 14.28, -8/7, -5/7\}$. The system can be represented in Lur'e form (3.85) by Eqs. (3.92) and (3.93). For $\tau > 0.038$ no feasible points were found such that Y (4.49) can be negative definite. The feedback matrix

$$G = \begin{bmatrix} 7.6909 & 2.1313 & -3.9865 & -0.3491 & 0.1811 & -0.5180 \\ -1.0520 & 0.0835 & 0.3455 & 8.0879 & 1.8021 & -4.8256 \end{bmatrix}$$

which is found for the maximum delay $\tau^* = 0.038$, has stabilized the error system for $\tau \in [0, 0.038]$. In the experiments

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

has been taken. In Figures 4.11(a)- 4.11(c) the result is given for the maximum obtained delay value $\tau^* = 0.038$. Simulation results for a smaller delay value $\tau = 0.01$ are shown in Figures 4.11(d)-4.11(f). In Matlab Simulink, synchronization has been observed until $\tau = 0.17$ for the same feedback matrix. Synchronization has not been observed when the delay was larger than 0.17. Figures 4.11(g)-4.11(i) show the result for $\tau = 0.18$. During the simulation, the initial conditions for the master and slave systems are taken as $x(0) = [-0.2 \ -0.2 \ -0.33 \ 0.2 \ 0.9 \ 0.33]^T$, $z(0) = [0.2 \ -0.2 \ 0.33 \ 0.2 \ -0.2 \ 0.33]^T$.

In [304], a mutual synchronization scheme for Lur'e systems with time delay has been investigated and Theorem 4.6 with slightly changed matrix inequality (4.49) has been used to obtain a sufficient condition for delay-dependent synchronization. The results for Chua's circuit are illustrated.

4.9.3 Experimental confirmation of nonlinear H_∞ synchronization for Chua's circuit and 5-scroll attractors

Nonlinear H_∞ synchronization with full static state error feedback and dynamic output feedback have been experimentally confirmed by computer simulation for Chua's circuit in [257] and [243], respectively. Here, we apply the nonlinear H_∞ synchronization method to a system that consists of two unidirectionally coupled generalized CNNs made of Chua's circuit cells (see Section 2.2.5.3) with an external reference input. The autonomous case (without external reference inputs) has been studied under the framework of master-slave synchronization in Section 4.9.1. Now, we study the same master-slave system with vector field

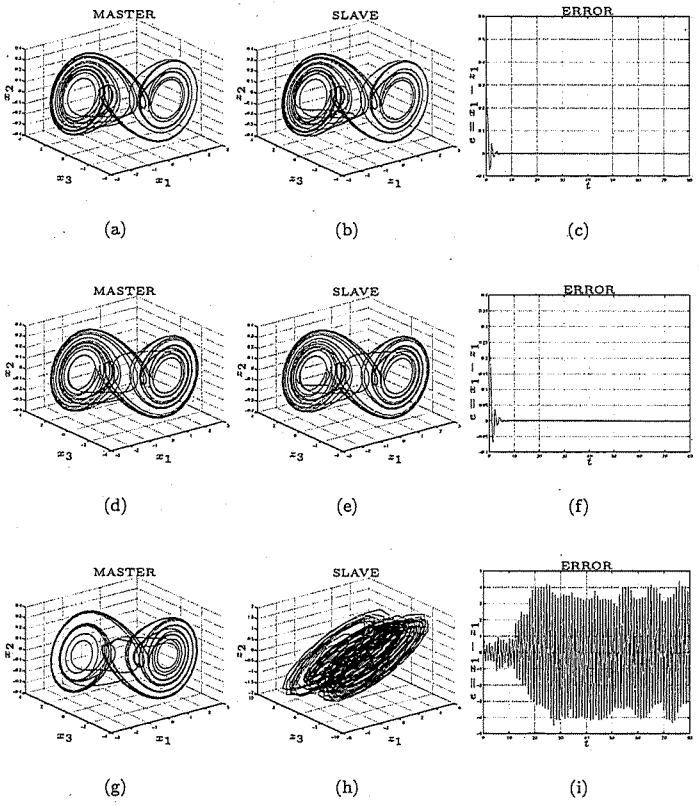


Figure 4.10: Simulation result for master-slave synchronization of two identical Chua's circuits. The master system is coupled to the slave system with first state variable and delay values $\tau = 0.039$, $\tau = 0.01$ and $\tau = 0.22$: (a), (d) and (g) Three-dimensional view on the double scroll attractor generated at the master system (b), (e) and (h) and the slave system; (c), (f) and (i) Error signal $(x_1(t) - z_1(t))$ with respect to time, respectively.

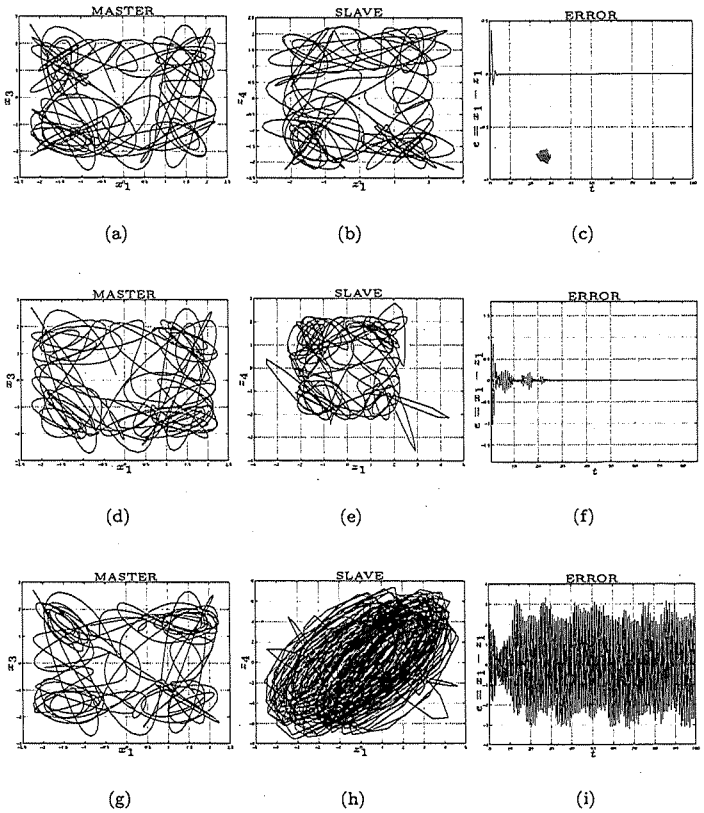


Figure 4.11: Simulation result for master-slave synchronization of two identical hyperchaotic systems. The master system is coupled to the slave system with first and fourth state variable and delay values $\tau = 0.038$, $\tau = 0.01$ and $\tau = 0.18$: (a), (d) and (g) 2-scroll hypercube attractor generated at the master system (projection onto the (x_1, x_4) plane) (b), (e) and (h) and the slave system; (c), (f) and (i) Error signal $(x_1(t) - z_1(t))$ with respect to time, respectively.

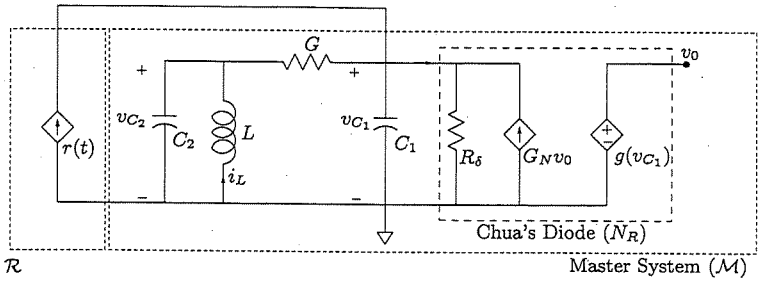


Figure 4.12: The master system which is driven by an external reference input for the nonlinear H_∞ synchronization. The slave system is shown in Figure 4.9.

modulation of the master system by the message signal (external reference input $r(t)$)

$$\begin{array}{rrl} \mathcal{R}: & d= & r \\ \mathcal{M}: & \dot{x}= & Ax+B\,\sigma(Cx)+Dd(t) \\ \mathcal{S}: & \dot{z}= & Az+B\,\sigma(Cz)+u \\ \mathcal{C}: & u= & F_\sigma(\sigma(Cx)-\sigma(Cz))+FH(x-z) \end{array} \tag{4.108}$$

with state vectors $x=[v_{C_1}\ v_{C_2}\ i_L]^T$ and $z=[\hat{v}_{C_1}\ \hat{v}_{C_2}\ \hat{i}_L]^T$ for the master $\mathcal{M}$ (see Figure 4.12) and controlled slave system $\mathcal{S}$ (see Figure 4.9), respectively. Here the synchronization scheme is examined for Chua's circuit and also a 5-scroll attractor from a generalized Chua's circuit. The matrices related to Chua's circuit and 5-scroll attractors in this scheme are given in Eq. (3.88) and Eq. (3.89) with their nonlinearities, respectively. The master system is driven by the message signal through the matrix $D=[1\ 0\ 0]^T$. The controller is characterized by the same H, F_σ matrices in Eq. (4.105) and $F=a_{01}\alpha\overline{m}_l$ where

$$l=\begin{cases} 1 & \text{for Chua's Circuit} \\ 5 & \text{for 5-scroll attractor.} \end{cases}$$

The parameter value a_{01} has to be considered as a design parameter for the nonlinear H_∞ synchronization method. The error signal $e=x-z$ satisfies the error system

$$\mathcal{E}:\ \dot{e}=(A-FH)e+(B-F_\sigma)\,\eta(Ce;z)+Dr(t) \tag{4.109}$$

where η satisfies the sector condition $[0,1]$ for Chua's circuit (also for the 5-scroll attractor). The synchronization scheme is represented in standard plant form with tracking error (4.64) where $d=r(t)$ and $\beta=[1\ 0\ 0]^T$. Now, we are

ready to use Theorem 4.8 which proves dissipativity with finite L_2-gain (4.67) where $w=r$ of the synchronization scheme (4.108) with respect to a quadratic storage function (4.66) where $\xi=e$. One should note that the scheme (4.108) does not contain a dynamic linear filter $\mathcal{R}$ and the noise signal ϵ, therefore the matrix inequality (4.68) has to be replaced by the following one

$$Z=\begin{bmatrix} A_\sigma^TP+PA_\sigma+C^TC & PB_\sigma+C^T\Lambda & PD-C^T \\ B_\sigma^TP+\Lambda C & -2\Lambda & 0 \\ D^TP-C & 0 & -\gamma^2+1 \end{bmatrix}<0 \tag{4.110}$$

where $A_\sigma=A-FH$ and $B_\sigma=B-F_\sigma$. Now, the controller of the scheme can be designed on the basis of this matrix inequality by solving the nonlinear optimization problem (4.101).

As a solution to (4.101) we obtain $a_{01}=0.99$ for Chua's circuit (3.3) with the parameters $\{18, 28.56, -8/7, -5/7\}$ and 5-scroll attractors from generalized Chua's circuit (3.31) with the parameters $\{18, 28.56, 3, [0.2911\ \ -0.5776\ \ 0.282\ \ -0.5777\ \ 0.3077], [0.5305\ 0.900\ \ 2.0526\ 3.1875\ 4.5]\}$ using sequential quadratic programming with *constr* in Matlab's optimization toolbox for random starting points.

In the realization, the synthesis of dependent sources based on the voltage controlled voltage sources approach have been used for realization of two identical Chua's circuits. The electronic realization of Chua's circuit has been given in Section 3.1.2. Like before, currents and time were scaled by a factor 10000. The current $r(t)$ driving the master system (see Figure 4.12) has been obtained as the current of a VCCS with a gain factor 10^{-4}S. The $(v_{C_1}(t), v_{C_2}(t))$ plot for the master system driven by a square wave $r(t)$ with $f=1kHz$ and the $(\hat{v}_{C_1}(t), \hat{v}_{C_2}(t))$ plot for the slave system are given in Figures 4.13(a) and 4.13(b), respectively. Figure 4.13(c) shows the synchronization between the master and slave systems. The horizontal axis corresponds to v_{C_1} of the master system and the vertical axis to $\hat{v}_{C_1}(t)$ of the slave system. The waveform of the message signal is shown at the top of Figure 4.13(d) and the first component of the error signal $s(t)=v_{C_1}(t)-\hat{v}_{C_1}(t)$ is displayed at the bottom of Figure 4.13(d). The message signal is binary valued. Therefore, the original message $r(t)$ is recovered then as

$$\hat{s}(t)=\text{sign}[s(t)]. \tag{4.111}$$

The message signal is shown in the middle of Figure 4.13(e) and the recovered message signal $(\hat{s}(t))$ is also displayed at the same figure with bigger amplitude which clearly shows that the message signal is recovered at the slave system. Figure 4.13(f) shows the influence of parameter mismatch between the master and slave systems for this scheme. Robustness of the scheme is examined with

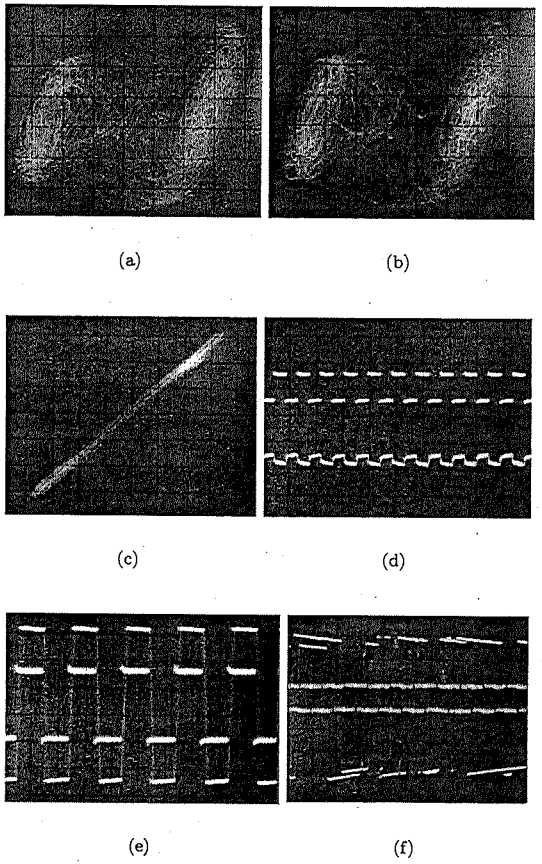


Figure 4.13: Nonlinear H_∞ synchronization using Chua's circuit: (a) $(v_{C_1}(t), v_{C_2}(t))$ for the master system $(0.5V/div - 0.2V/div)$, (b) $(\hat{v}_{C_1}(t), \hat{v}_{C_2}(t))$ for the slave system $(0.5V/div - 0.2V/div)$, (c) the plot of $v_{C_1}(t)$ versus $\hat{v}_{C_1}(t)$ $(0.5V/div - 0.5V/div)$, (d) the message signal $r(t)$ and the first component of the error signal $s(t) = v_{C_1}(t) - \hat{v}_{C_1}(t)$ $(0.1V/div - 50mV/div)$, (e) $r(t)$ and recovered message $\hat{s}(t)$ using comparator $(5V/div - 20mV/div)$, (f) desynchronization in the H_∞ sense for $R_\delta = 9k\Omega$ $(5V/div - 20mV/div)$.

respect to sensitivity of the R_δ resistance value. The results of the recovered message signal from the nonlinear H_∞ synchronization scheme are shown in Figure 4.13(e) and Figure 4.13(f) for the case of identical ($R_\delta = 10k\Omega$) and non-identical (the R_δ resistances values of the master and slave systems are taken $10k\Omega$ and $9k\Omega$, respectively) master and slave systems, respectively. Figure 4.13(f) shows the desynchronization in the H_∞ sense.

The nonlinear H_∞ synchronization scheme is also confirmed for 5-scroll attractors from generalized Chua's circuit. For 5-scroll attractors, a realization of the master and controlled slave systems using ideal circuit elements are also shown in Figures 4.12 and 4.9. In order to realize 5-scroll attractors, the breakpoints of the PWL characteristic (3.37) are scaled with $k = 2$ (which was the subject of Section 3.2.1.2). The measured values of the scaled breakpoints were given above when a 5-scroll attractor is observed in the master system (Figure 4.14(a)). The realization of 5-scroll attractors has been given in Section 3.2.1.1.

Similar observations have been obtained when Chua's circuits were replaced by 5-scroll attractors. The $(v_{C_1}(t), v_{C_2}(t))$ plot for the master system driven by a square wave $r(t)$ with $f = 11kHz$ and the $(\hat{v}_{C_1}(t), \hat{v}_{C_2}(t))$ plot for the slave system are given in Figures 4.14(a) and 4.14(b), respectively. Figure 4.14(c) shows the synchronization between the master and slave systems. The waveform of the message signal is shown at the bottom of Figure 4.13(d) and the first component of the error signal $s(t) = v_{C_1}(t) - \hat{v}_{C_1}(t)$ is displayed at the top of Figure 4.14(d). The message signal is shown at the bottom of Figure 4.14(e) and the recovered message signal $(\hat{s}(t))$ is displayed at the top of Figure 4.14(e) which clearly shows that the message signal is recovered at the slave system. Robustness of the scheme for 5-scroll attractors is also examined with respect to sensitivity of the R_δ resistance value. The results of the recovered message signal are shown in Figure 4.14(e) and Figure 4.14(f) for the case of identical ($R_\delta = 10k\Omega$) and non-identical (the R_δ resistances values of the master and slave systems are taken $10k\Omega$ and $9k\Omega$, respectively) master and slave systems, respectively.

4.10 Conclusion

One important historical achievement in nonlinear dynamics was the synchronization of chaotic systems without loss of their chaotic nature [110]. This achievement makes it possible to recover the information signal which is carried by a chaotic signal. In this chapter an overview has been given of several methods for chaotic systems synchronization for the class of Lur'e systems, including full static state feedback, measurement feedback schemes, robust syn-

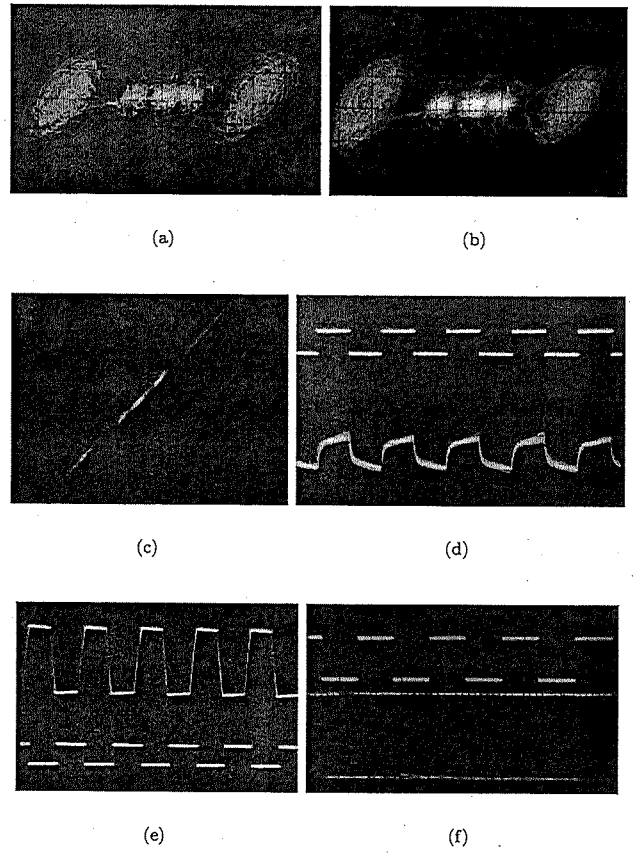


Figure 4.14: Nonlinear H_∞ synchronization using 5-scroll attractors: (a) $(v_{C_1}(t),v_{C_2}(t))$ for the master system $(0.2V/div - 50mV/div)$, (b) $(\hat{v}_{C_1}(t),\hat{v}_{C_2}(t))$ for the slave system $(0.2V/div - 50mV/div)$, (c) the plot of $v_{C_1}(t)$ versus $\hat{v}_{C_1}(t)$ $(0.2V/div - 0.2V/div)$, (d) the message signal $r(t)$ and the first component of the error signal $s(t) = v_{C_1}(t) - \hat{v}_{C_1}(t)$ $(20mV/div - 50\mu sn/div)$ and $(20mV/div - 50\mu sn/div)$, (e) $r(t)$ and recovered message $\hat{s}(t)$ using comparator $(20mV/div - 50\mu sn/div$ and $1V/div - 50\mu sn/div)$ (f) desynchronization in the H_∞ sense for $R_\delta = 9k\Omega$ $(1V/div - 50\mu sn/div)$.

chronization methods, autonomous and non-autonomous schemes, synchronization schemes with time delays and impulsive synchronization methods. While in the past many of such schemes were investigated in a rather ad hoc fashion, presently several systems and control theoretical interpretations have been given to synchronization methods. Especially, the nonlinear H_∞ synchronization scheme provides the rigorous mathematical proof for recovering the information signal from chaotic carrier. A considerable advantage for the class of Lur'e systems is that it is straightforward to derive sufficient conditions for global synchronization which can be expressed in terms of matrix inequalities. Many circuits and systems of practical interest such as Chua's circuit and generalized chaotic and hyperchaotic versions, were represented in this Lur'e form in the previous Chapter and they were directly applicable within the synchronization schemes explained in this Chapter.

Chapter 5

Engineering Applications

So far our concerns were to provide the reader with different theoretical foundation and frameworks. Now we will move towards applications of the subjects in engineering. In Section 5.1, chaos communication, which is probably the first application of chaos in engineering, is considered to improve the performance with the presented synchronization methods of Chapter 4. In Section 5.2, we study a relatively new application of chaotic systems and signals in optimization. In this Section we also combine the ideas of the previous Chapters 2, 3 and 4 to improve the performance of an optimization network. In Section 5.3, we address a most encouraging application of chaotic signals in engineering which is its use as a source for random number generators and we propose a novel true random bit generator (TRBG) based on a double scroll attractor. In the same Section, we also show how the CNN-UM can be used to generate a binary pseudo random pattern for image and video authentication which is studied in Section 5.4. In Section 5.5, we describe a solution to find templates in order to make the CNN chip react as an ideal CNN structure. New contributions are made in Sections 5.2.1, 5.2.2, 5.3.1, 5.3.2, 5.4.2 and 5.4.3.

5.1 Chaos in communications

The pioneering work on chaos synchronization [197] has led to employing chaos in communications applications. Chaos has potential applications in several functional blocks of a communication system (see Figure 5.1) such as encryption, decryption [138, 65], modulation and demodulation [132, 1]. Three particular properties of chaos which are a continuous spectrum, a complex structure

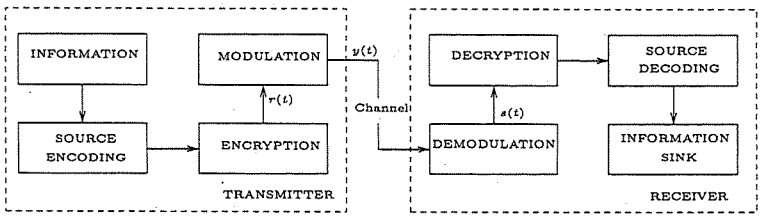


Figure 5.1: General structure of a communication system.

and aperiodic behavior have been individually exploited in spread-spectrum, secure and multiuser communication applications, respectively [1].

In this Section, we focus on modulation and demodulation blocks of the communication system (Figure 5.1). Modulation is a process by which a signal is transformed into a chaotic waveform (carrier) using a chaotic system and then sent through the channel. At the receiver, this signal may be recovered by means of either *coherent* or *noncoherent* detection. In chaos communication, coherent detection is accomplished by synchronizing the chaotic systems of the transmitter with its copy at the receiver. In noncoherent detection (which is for digital communication), synchronization is not required. In this case, the receiver estimates the properties of chaotic signals which carry the symbols. The role of synchronization in digital communication has been extensively studied by Kolumban *et al.* [147, 148, 146] who have stated that synchronization based coherent receivers have advantages over noncoherent receivers in terms of bandwidth efficiency, data rate and noise performance. Furthermore they reported that these advantages are lost if synchronization cannot be preserved [147, 148, 146]. Therefore, synchronization is an important issue in order to have better performance than conventional communication systems. Here, we concentrate on the possible usage and advantage of the presented synchronization methods of Chapter 4 for communication applications.

5.1.1 Modulation and demodulation based on chaos

Since the observation of synchronization, various modulation and demodulation schemes have been proposed for chaos communications. The most common chaos communication methods are classified into two main categories based on the transmitter structure, namely, *static modulation* and *dynamic modulation* methods [1].

5.1.1.1 Static modulation methods

Static modulation methods [1] are like the standard modulation methods: amplitude modulation and phase modulation. In these methods the message carrier $y(t)$, the chaotic waveform, is independent of the message itself. Chaotic masking [139] is a static modulation method, which is the first chaos communication system. In chaotic masking, the information signal $r(t)$ is spread by adding it to the output $x(t)$ of a chaotic system. The resulting signal $y(t)$

$$y(t) = x(t) + r(t) \tag{5.1}$$

is modulated and transmitted. An identical chaotic system in the receiver can be made to synchronize with $x(t)$. This allows the receiver to recover the message signal $r(t)$ (the recovered signal will be denoted by $s(t)$). Chaotic masking is basically an amplitude modulation scheme and suffers from the disadvantage that noise introduced by the channel is indistinguishable form the message signal. Therefore, this scheme has became unreasonable for any communication applications. Chaotic masking can be exploited in spread-spectrum communication applications.

Chaos Shift Keying (CSK) [67] is also a static modulation method designed for digital communication. An information signal is encoded by transmitting one chaotic signal for a "1" and another chaotic signal when a binary "0" needs to be transmitted and the modulation of CSK is described by

$$y(t) = \begin{cases} x_1(t), & \text{if} \quad r(t) = 0 \\ x_2(t), & \text{if} \quad r(t) = 1. \end{cases} \tag{5.2}$$

For the CSK method, two demodulation schemes are possible: coherent and noncoherent. The coherent receiver contains copies of the systems corresponding to "0" and "1". One of these will synchronize with the incoming signal and the other one will desynchronize depending on the transmitted signal. Differential chaos shift keying [151], chaotic on-off keying [149] and frequency modulation differential chaos shift keying [150] are variants of CSK. CSK is exploited in spread-spectrum and multiuser communication applications.

5.1.1.2 Dynamic modulation methods

In dynamic modulation, the message carrier is a function of the message. Chaotic modulation [103, 250] is a dynamic modulation method which is described by

$$\begin{aligned} \dot{x} &= f(x, r(t)) \\ y &= g(x, r(t)). \end{aligned} \tag{5.3}$$

Chaotic modulation schemes are applicable to digital and analog messages. When one compares chaotic modulation with chaotic masking, one can recognize that noise introduced by the channel does not directly affect the message which is modulated by the chaotic modulation scheme itself.

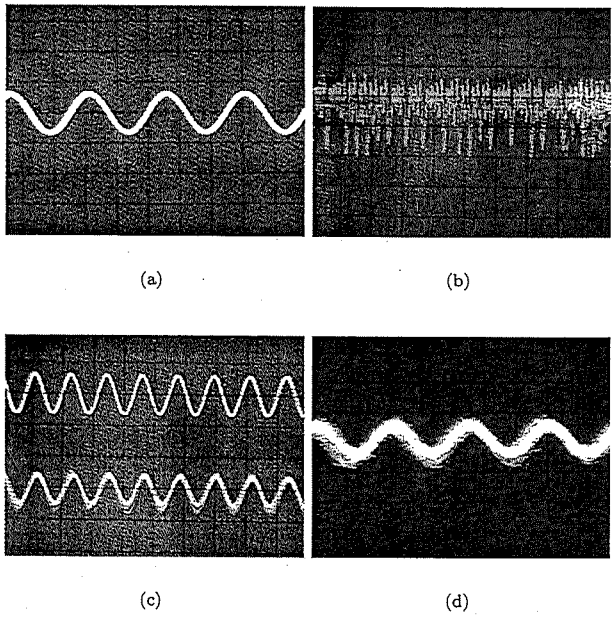
In fact, the chaotic modulation scheme has been already studied under the nonlinear H_∞ synchronization method explained in Section 4.5. The receiver is designed based on nonlinear H_∞ synchronization which provides the rigorous mathematical proof for recovering the message signal in the chaotic carrier $y(t)$. Experimental confirmations of chaotic modulation and demodulation based on the nonlinear H_∞ synchronization has been already given for Chua's circuit and 5-scroll attractor in Section 4.9.3. Figures 4.13(d) and 4.14(d) shows a digital message signal $r(t)$ (at the top) and recovered message signals $s(t)$ (at the bottom) when the message signals are modulated by Chua's circuit and 5-scroll attractors, respectively. Figures 4.13(e) and 4.14(e) show that the recovered messages $\hat{s}(t)$ after the thresholding the signal $s(t)$ by Eq. (4.111) are able recover the digital information.

Experimental results have been presented in [227] when an analog message is modulated by chaotic modulation. The scheme which is given in [227] has been analyzed under the nonlinear H_∞ synchronization method in [300] for Chua's circuit. The same experiments can be repeated with the set-up of Section 4.9.3. Figure 5.2(a) shows the waveform of the message $r(t)$ which is used in that experiment [227]. The chaotic carrier after the message $r(t)$ in Figure 5.2(a) is shown in Figure 5.2(b). The message signal and a demodulated signal are shown at the top and bottom of Figure 5.2(c), respectively. Figure 5.2(c) clearly shows that the message signal is recovered at the receiver side for an ideal channel. In order to examine the affect of noise on the channel, white noise which is supplied by an analog noise generator is added to the chaotic carrier signal $y(t)$. The observed waveform for recovered signal $s(t)$ is given in Figure 5.2(d) for the same message signal in Figure 5.2(a). In that experiment [227], the message signal can be recovered using the simple master-slave synchronization scheme for the maximum %23 noise/signal ration.

5.1.2 The role of new synchronization schemes

While the nonlinear H_∞ synchronization and its robust version against the parameter mismatch (Section 4.6) can be directly used in dynamical modulation methods, the rest of the synchronization scheme in Section 4 can be employed in static modulation methods.

In the first part [147] of the extensive studies of Kolumban *et al.* [147, 148, 146], it has been stated that synchronization of chaos is not sufficiently robust for



(a) (b)

(c) (d)

Figure 5.2: (a). The message signal ($5\,V/div - 20\mu sn/div$). (b). The chaotic carrier ($5\,V/div - 2msn/div$). (c). The message signal $r(t)$ (the signal above) ($5\,V/div - 50\mu sn/div$), the recovered message $s(t)$ ($1V/div - 50\mu sn/div$) . (d). The recovered message $s(t)$ under the channel noise affect ($0.5\,V/div - 20\mu sn/div$).

practical applications in communications. Therefore, they suggested to use noncoherent based chaos communication. In fact, their statement is based on the experimental results of synchronization for the Pecora-Carroll scheme. They have reported that synchronization should be robust against the effect of parameter mismatch, sensitivity to noise and poor propagation. Since the time these findings were reported in [148], a large variety of new synchronization schemes have been developed for different purposes. For instance the robust synchronization (Section 4.3) and time-delay synchronization (Section 4.4) schemes are designed to improve the synchronization against a parameter mismatch and a propagation delay on the channel. Hence, the Pecora-Carroll

Furthermore, Kolumban *et al.* [148] have also reported that in the Pecora-Carroll scheme, synchronization is valid only for trajectories in the receiver that come sufficiently close to the reference trajectory in the transmitter (the reason is that the synchronization criterion is a local condition). Therefore, synchronization is not reached until two trajectories come close to each other. In a practical communication system, due to the channel noise it is not guaranteed that these two trajectory remain close to each other [148]. However, synchronization criteria for master-slave schemes (in Chapter 4) are global conditions.

As a conclusion, synchronization can be robustly preserved with new methods like in Section 4 so that the advantages of synchronization which were mentioned in the first statement of Kolumban *et al.* [147, 148, 146] are kept. Furthermore, in order to use these synchronization schemes in chaos communication methods one can find possible chaotic systems and their circuit implementations according to Chapter 3.

5.2 Chaotic systems in optimization

Neural networks have been used to solve optimization problems since Hopfield and Tank [114] proposed their recurrent network. While the Hopfield network provided a new tool to solve a variety of optimization problems, the steepest descent dynamics nature of the method often lead to local minima of the energy landscape, which limits the network's potential for solving more complex or larger scale problems.

In order to alleviate the shortcomings of Hopfield networks and also the steepest descent method itself, chaotic dynamics and an ensemble of steepest descent algorithms (see Section 2.4.2) have been introduced by various researchers, respectively. There are two approaches for studying the role of chaotic dynamics in the Hopfield network for solving combinatorial optimization problems. The

first one is the *internal* approach where the chaotic dynamics are generated internally by the optimization network. In the second approach, the *external* one, the chaotic dynamics are externally generated by another network and is injected into the optimization network [155].

Chaotic simulated annealing is probably the first internal approach as introduced by Chen and Aihara [35]. In order to make use of the advantage of chaotic dynamics, Chen and Aihara have developed a *transiently chaotic neural network* (TCNN) for combinatorial optimization problems, by introducing a new variable corresponding to the temperature in the usual annealing process into a chaotic neural network model [2]. In fact, the temperature corresponds to a bifurcation parameter that drives the dynamics of the network from a chaotic phase to a stable phase. When the network reaches a stable phase, it approaches a dynamical structure similar to Hopfield network networks. Another chaotic simulated annealing approach based on varying the timestep of an Euler discretized Hopfield network has been developed by Wang and Smith [279]. In this case, the timestep acts as the bifurcation parameter.

External approaches mainly use chaotic signals as a rich perturbation signal with noise like properties. In [111] Hayakawa *et al.* reported a neural network driven by a chaotic signal (from the logistic map) operating efficiently to solve a travelling salesman problem (TSP). Meanwhile Asai [13] *et al.* injected various types of chaotic signal into Hopfield models. They have found that the network state wanders frequently among the local minima when the correlated noises from chaotic maps are injected into the network. They have been followed by Hasegawa *et al.* [108] who showed that noises having the same autocorrelation as the chaotic noise offer similar improvements to solve the TSP. Recently, He [112] has introduced a new external approach which decays exponentially the external chaotic noise until it vanishes. He reported that the quality is higher than that of external and internal approaches proposed in [111] and [155], respectively.

In order to overcome the problem of unwanted local minima of gradient networks (Eq. (2.47)) and CLMs (Eq. (2.52)), we study the use of chaotic signals in the following Sections.

5.2.1 Chaotic annealing

In the Chaotic Annealing (CAn) method studied within this Section, a continuous time optimization algorithm of the following form is considered:

$$\dot{\xi} = -\mu \nabla_{\xi} U(\xi) + T d(t)\, e^{-\tau t} \tag{5.4}$$

with $T \in \mathbb{R}$. $d(t) \in \mathbb{R}^n$ is an external driven force vector taken as output from a 1-D CNN consisting of identical n-scroll attractors with unidirectional coupling between the cells. This 1-D CNN produces a hyperchaotic attractor (see Section 3.4.2). The approach is similar to the use of noise in continuous simulated annealing [88] where the noise (within a stochastic differential equation) is replaced by deterministic chaos. The value τ is a cooling speed time constant. The network is initially forced by a hyperchaotic signal and eventually this external signal vanishes. In this Section, the hyperchaotic attractor is obtained from a n-scroll hypercube attractor Eq. (3.80) with L the number of cells in the one-dimensional CNN array. We take the Chua circuit (2-scroll attractor) with the parameters $\{9, 14.286, -1/7, 2/7\}$ and $k = 0.06$. Here $d(t) = [x_1^T\ x_2^T\ \cdots x_L^T]^T$ is taken as driving force for the optimization algorithm. The equations (5.4) and (3.80) are simulated simultaneously.

Chaotic annealing is also an external approach like [111, 13, 108, 112]. Here, the proposed method is aimed at improving the gradient system instead of the Hopfield network for combinatorial optimization problems. Furthermore, the method uses a hyperchaotic signal as a perturbation signal such that the network can explore the search space in any direction without repeating itself.

5.2.2 Coupled chaotic annealing

Instead of taking steepest descent gradient systems (2.47) within the CLMs (2.52), we now couple different chaotic annealing systems (5.4) within a CLM:

$$\begin{cases} \dot{\xi}_i &= -\frac{\eta}{q}\nabla_{\xi_i} U[\xi_i] + T d_i(t)\, e^{-\tau t} + \gamma_{i-1}[\xi_{i-1} - \xi_i] \\ & \quad -\gamma_i[\xi_i - \xi_{i+1}] + \lambda_{i-1} - \lambda_i \\ \dot{\lambda}_i &= \xi_i - \xi_{i+1} \end{cases} \tag{5.5}$$

where $d_i(t) \in \mathbb{R}^n$, $i \in 1, 2, ..., q$ and $L = q$ and the hypercube attractors are simulated simultaneously with the CLM dynamics. Coupled chaotic annealing (CCA) integrates two approaches which prevent the network from being trapped at the local minima, CLM and CAn. While each minimizer of the network fluctuates according to the hypercube attractor and the cooling, simultaneously each minimizer also exchanges the information between the perturbed local search process. Here the hypercube attractor plays an important role in exploring the search space, which makes sure that each minimizer is perturbed by a signal which never repeats itself and expands in different directions for each minimizer. Hence, each minimizer explores a different part of the search space.

CCA can also be interpreted within the CNN framework. CCA can be thought then as two unidirectionally coupled 1-D CNNs which work cooperatively to-

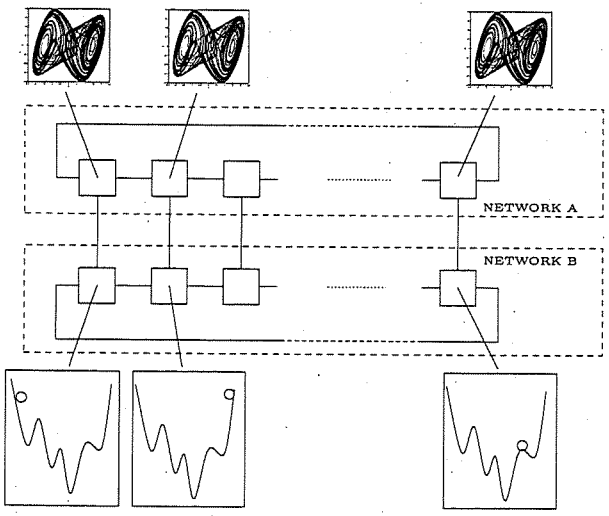


Figure 5.3: CNN interpretation of coupled chaotic annealing. Network A drives network B with chaos. Network B performs optimization with cooperative search (using coupled cells of optimization processes).

wards finding improved local minima (Figure 5.3). In this cooperation, while one part of the network produces the hyperchaotic signal, the other part processes information together with the output of the first network in such a way that the cost function is minimized. The process is ended when state synchronization is achieved [259].

5.2.3 Illustrative examples

We illustrate and compare the methods on a number of examples. A first test function (2.55) (see Example 2.1) with $a = 0.01$, $\omega = 0.2$ has been optimized here for $n = 3$ and $n = 10$ with random initial conditions in $[-10000, 10000]^n$. This test function has been investigated in [239] for simulated annealing and stochastic approximation methods. For the annealing scheme was taken $T = 50$, $\tau = 0.01$, $\mu = 65$ (case $n = 3$), $T = 150$, $\tau = 0.01$, $\mu = 20$ (case $n = 10$), and $\mu = 65$, $\gamma_i = 0.01$ $\forall i$. A number of $q = 3$ minimizers has been taken and 100 different runs with random initial conditions were taken. The equations were simulated in Matlab using ode23 (in order to avoid storing state variables for all time instants the Matlab option [t,x] = ode23('filename', tspan, x0) is used where tspan specifies the time instants to be returned). Boxplots of the simulation results with 100 runs are shown in Figure 5.4. Each time, uncoupled gradient networks (GN), CLM, CAn and CCA are compared for the same set of initial conditions. One observes a significant improvement on these test functions of CLM, CAn, CCA w.r.t. GN. The use of chaotic annealing leads to improvements in performance and CCA leads to the best variance reduction on the results.

In Figure 5.5 results of the method are shown on a 10-city Travelling Salesman Problem (TSP) problem. Let N be the number of cities and $d_{x,y}$ be distance between cities x and y. A tour of the TSP can then be represented by a $N \times N$ permutation matrix [114], where each row and column is associated with a particular city and order of visiting the cities within the tour, respectively. As cost function is taken here

$$\begin{aligned} U(v) = & \frac{A}{2} \sum_x \sum_i \sum_{j \neq j} v_{x,i} v_{x,j} + \frac{B}{2} \sum_i \sum_x \sum_{y \neq x} v_{x,i} v_{x,j} \\ & + \frac{C}{2} \left(\sum_x \sum_i v_{x,i} - N \right)^2 + \frac{D}{2} \sum_x \sum_{y \neq x} \sum_j d_{x,y} v_{x,i} (v_{y,i+1} + v_{y,i-1}) \end{aligned} \tag{5.6}$$

where $v_{x,i}$ represents visiting city x as i-th city and the indices $i+1$ and $i-1$ are taken modulo N. This has been analyzed and proposed in the area of Hopfield networks [114]. We consider here implementations which are different from the Hopfield network (dynamics in state variables $u_{x,i}$ with $v_{x,i} = \frac{1}{2}(1+\tanh(u_{x,i}))$)

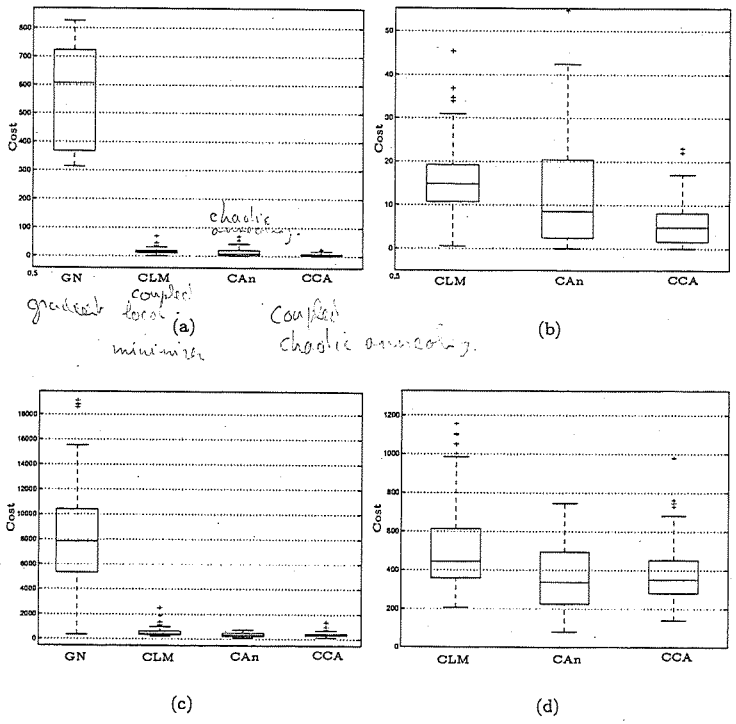


Figure 5.4: Test function for $n = 3$ and $n = 10$: (a,c) comparison between GN, CLM, CAn, CCA (from left to right); (b,c) enlarged top-figure for CLM, CAn, CCA, respectively.

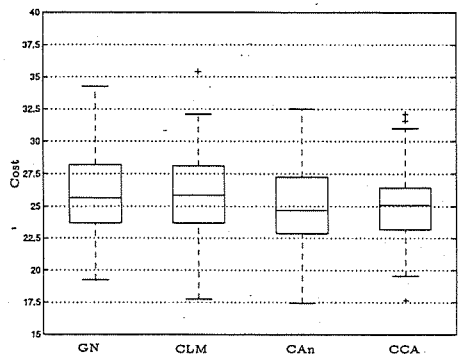


Figure 5.5: 10-city TSP problem: comparison between GN, CLM, CAn, CCA.

for comparison reasons (comparisons made between GN, CLM, CAn, CCA). The simulation results for 100 different runs and $D = 1, C = 30, A = 20, B = 20$, $T = 0.4$, $\tau = 0.5$, $\gamma = 40$, $q = 5$, $\mu = 5$ with random initial conditions indicate a good performance with low variance of the CCA technique (Figure 5.5).

5.3 Random number generators and cryptography

Random number generators are used in many areas including computer simulations, Monte-Carlo techniques in numerical analysis, test problem generation for the performance evaluation of computer algorithms, statistical sampling, stochastic optimization methods (such as simulated annealing), watermarking for image authentication and cryptography. A good random number generation helps to improve the results in these applications and is vital for cryptographic security. Security issues are coming to the fore today because of increasingly demanding security requirements in many new applications on the internet (such as secure e-mail, e-commerce, e-government). The security of cryptographic algorithms depends on generating secret quantities which are generated by random number generators, for passwords, keys, etc. As noted in [73], the use of pseudorandom processes to generate secret quantities can result in pseudo-security. Random numbers can be generated by a random bit generator which

can be defined as a device or algorithm whose output is a sequence of statistically independent and unbiased binary digits [179]. To ensure that a random number generator is secure, its output must be statistically indistinguishable from a true random sequence and unpredictable [179, 73].

Random number generators (RNGs) can be classified into three classes which are *true random number generators (TRNGs)*, *pseudorandom number generators (PRNGs)* and *hybrid random number generators (HRNGs)*. A true random number generator operates by measuring unpredictable natural processes such as thermal noise from a semiconductor [52], frequency instability of an oscillator [81], elapsed time between emission of particles during radioactive decay [98] and variations in disk drive response times [66, 116]. These are also called hardware-based generators because of the use the randomness aspect in the hardware. The processes are chaotic or non-deterministic. Furthermore, they produce continuous time analog signals which are often called noise. As is shown in [15, 169] true random bits/numbers can be generated by these physical noise sources.

Pseudorandom number generators use deterministic processes (also called *deterministic random number generators*) to generate a series of outputs from an initial seed state. PRNGs are much more cost-effective and thousands of times faster than hardware based random number generators. However, because the output is a function of the seed state, the actual entropy of the output can never exceed the entropy of the seed [118, 228]. Hence, the randomness level of the pseudorandom numbers depends on the level of randomness of the seed. Therefore, hybrid random number generators use a random generator as a seed generator and expand it. A seed generator is a hardware-based RNG with or without user's interaction such as random keystrokes, mouse movements or hard drive seek times.

If one summarizes the present state-of-the-art in random number generators, the conclusion is that pseudorandom number generators need a true random number generator in order to improve the quality of the output (e.g. the hardware-based Intel Random Number Generator [52] included in the Intel 8XX series of chip-sets for generating the random seed [229]). Although commercial cryptographic packages have user's interaction to produce a supposedly random seed and do not need additional hardware, these methods are time consuming, inconvenient for the user and cannot be used with automated scripts [116, 118]. Nevertheless, it is quite difficult to distinguish between a signal coming from a non-deterministic source and a chaotic system. In this thesis, we group true random number generators into two classes, based on the kind of analog signal they are using. It will be called a true random number generator either based on a chaotic or on a non-deterministic system (e.g. while the

sources of [52, 81, 98] are known as a non-deterministic system, variations in disk drive response time are known to be a consequence of chaotic air turbulence [81]). The same classification of TRNGs is also made in [169].

The output of chaotic systems can be predicted from the exact initial conditions depending on the first Lyapunov exponent. However, sensitivity with respect to the initial conditions causes unpredictability on the long term. The short term predictability of chaotic time-series was shown in [72, 25] and has been also addressed in the time-series prediction competition reported in [248]. Nevertheless, short term predictability of chaotic time-series is seen as a drawback for using chaotic signals in random number generators which are designed for cryptographic applications. A scheme which is taking into account the *predictability time* is studied in [20]. In [94] a non-invertible function is used as a static output of a chaotic system which also improves the randomness for the output signal. In fact, non-invertibility (also called a *one-way function* in cryptography) is very much used in many areas in cryptography such as public key cryptography, authentication [179] and also for generating pseudorandom numbers [233, 179]. Similar ideas have been used by Stojanovski and Kocarev [237] who define a binary generating partition which can be considered as a one-way function, on the chaotic piecewise-linear one-dimensional map for producing random bits.

One should note that when unpredictability is an important issue, direct use of the output of the chaotic system might give an advantage to the opponent if it is short term predictable. Cryptographic protocols crucially depend on the unpredictability of the secret key which is produced by a random number generator. If an attacker can predict the key's value or the number of keys, the protocol can be broken with much less effort than when truly random keys are used. Therefore, it is vital that the secret keys are generated from a true random source. Statistical tests determine whether the sequence possesses a certain attribute that a truly random sequence would be likely to exhibit. Hence any random number generator which is proposed for use in a cryptographic protocol, must be subject to statistical tests. There are two very-well known statistical test suites: FIPS-140-1 [179] and the Diehard test suite [172]. FIPS-140-1 is issued by the National Institute of Standards and Technology (NIST) and strong statistical tests are given in the Diehard test suite.

5.3.1 TRNG from a double scroll attractor

The aim of this Section is to show the usage of continuous-time chaotic systems for generating true random bits. We introduce a straightforward method for generating true random bits and take care of randomness in the sense of unpredictability. Due to the difficulty of proving unpredictability in a theo-

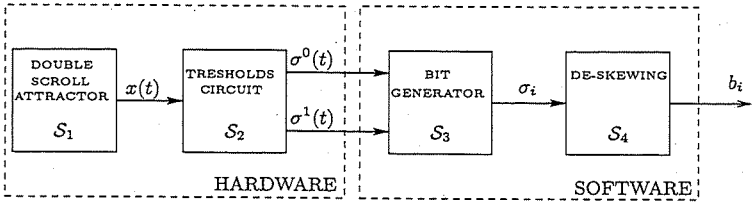


Figure 5.6: Proposed true random bit generator from a double scroll attractor. The proposed TRBG consists of a hardware and software parts. In the hardware part, the double scroll attractor is generated and the state space of the attractor is partitioned into three subspaces by the blocks $\mathcal{S}_1$ and $\mathcal{S}_2$, respectively. In the software part, while the block $\mathcal{S}_3$ is coding the scrolls, the block $\mathcal{S}_4$ is eliminating the bias in the output of the bit generator.

retical way, the proposed true random bit generator is subjected to statistical tests. We demonstrate that the proposed true random bit generator passes the FIPS-140-1 statistical tests and the full suite of the Diehard test. However, one should notice that none of the statistical tests proves that the generator is an ideal random number generator (the many tests are in fact necessary conditions but strictly speaking not sufficient).

Here we introduce a new True Random Bit Generator (TRBG) based on a chaotic system. Figure 5.6 shows the proposed TRBG which consists of a hardware and software parts. The software part can also be integrated in the hardware part using standard digital circuits. For practical reasons, this block is implemented here in software. The hardware part includes a chaotic oscillator and a thresholds circuit.

The chaotic oscillator $\mathcal{S}_1$ is a continuous time third order autonomous system which is described by Eqs. (3.56) and (3.58). We use $M_x = 1$ and $N_x = 0$ in Eq. (3.58) to obtain a double scroll attractor. We have implemented a double scroll attractor from the circuit in Figure 3.27. In order to obtain a double scroll attractor in the x state variable direction, we have removed the sub-circuits f_y and f_z and one comparator from the sub-circuit f_x. The passive component values are taken as $C_1 = C_2 = C_3 = 1nF$, $R_1 = 5.1k$, $R_2 = 7k\Omega$, $R_4 = 11k\Omega$ and $R_{x1} = 74.8k\Omega$. On the circuit, V_{x1} is measured as -0.51 Volt which has been obtained using series connected resistances between ground and -15 Volt. The corresponding result for these value is shown in Figure 5.7.

The state space of the double scroll attractor is partitioned into three subspaces by block $\mathcal{S}_2$ of the hardware part. Two of the subspaces are related with the

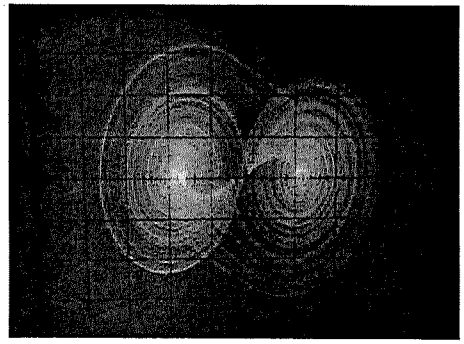


Figure 5.7: Double scroll attractor which has been used for the proposed TRBG.

location of the scrolls in state space. The third subspace is a region where the transition between the scrolls occurs. The bit generator (block $\mathcal{S}_3$) is the first block in the software part. Its inputs correspond to the outputs of block $\mathcal{S}_2$. The block $\mathcal{S}_3$ is basically coding the scrolls (e.g. the output of the bit generator is 0110 which means that the state space trajectory moved from scroll 0 to the other scroll 1, jumps to itself and then returns to scroll 0). In fact, blocks $\mathcal{S}_2$ and $\mathcal{S}_3$ are discretizing the state variable $x(t)$ in space instead of in time (relying on the ergodicity property of chaos). When the blocks $\mathcal{S}_2$ and $\mathcal{S}_3$ are combined and considered as one block its function is non-invertible. Sampling the chaotic signal in space gives an irregular sampling of the signal in time (Figure 5.8). Therefore, it becomes very hard to make a prediction for an opponent in cryptography, both a one-step ahead and long term prediction. In this way, the method presented in this thesis is a combination of the two ideas mentioned in the introduction and improves the randomness in the sense of unpredictability [94, 20].

The last block ($\mathcal{S}_4$) in the software part is a de-skewing which is eliminating the bias in the output of the bit generator. This bias depends on the probability density of the chaotic signal itself. Moreover, offset and bandwidth limitations might also effect the bias of the generated sequence [15].

5.3.1.1 Bit generation

A simple way to generate a bit sequence from a chaotic real valued signal is by using a threshold function [144]

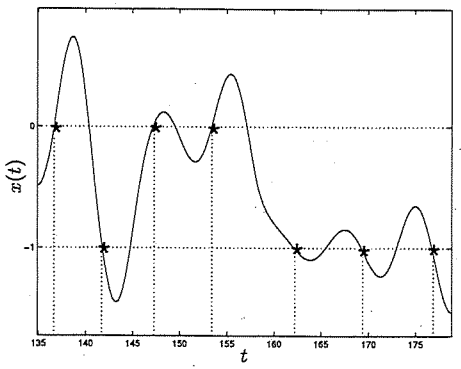


Figure 5.8: The state variable $x(t)$ is sampled in the space instead of in time. Sampling the chaotic signal in space gives an irregular sampling of the signal in time ($*$ shows the sampled points).

$$\sigma_c(x) = \begin{cases} 0 & \text{if } x < c \\ 1 & \text{if } x \geq c. \end{cases} \tag{5.7}$$

One obtains binary sequences which are referred to as chaotic binary sequences [144]. Because our chaotic system $\mathcal{S}_1$ generates a continuous time signal, we define two thresholds (c_1 and c_2). Then the state space is divided into three subspaces denoted by

$$V_0 = \{(x, y, z) | x \leq c_2\}, \tag{5.8}$$

$$V_T = \{(x, y, z) | c_2 < x < c_1\} \tag{5.9}$$

and

$$V_1 = \{(x, y, z) | x \geq c_1\}. \tag{5.10}$$

Figure 5.9 shows the subspaces for the values $c_1 = 0$ and $c_2 = -1$. The block $\mathcal{S}_2$ that realizes the two threshold functions, is described by

$$\mathcal{S}_2 : \quad \begin{array}{rcl} \sigma^1(x(t)) & = & \begin{cases} 0 & \text{if } x(t) < c_1 \\ 1 & \text{if } x(t) \geq c_1 \end{cases} \\ \sigma^0(x(t)) & = & \begin{cases} 0 & \text{if } x(t) > c_2 \\ 1 & \text{if } x(t) \leq c_2. \end{cases} \end{array} \tag{5.11}$$

By discretizing the state space in this way, one can obtain symbolic dynamics

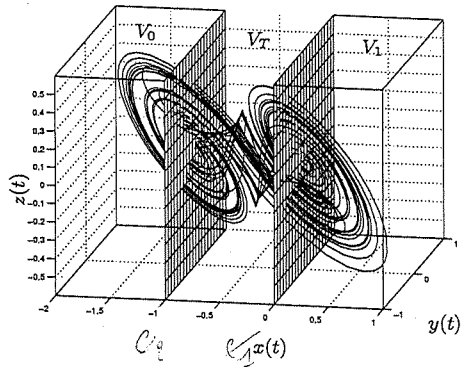


Figure 5.9: Discretizing the state space for $c_1 = 0$ and $c_2 = -1$.

that describe the evolution of the system by an infinite sequence of symbols [164, 97]. The two threshold functions are realized by LM311 type comparators (Figure 5.10) which are of the same type as the comparators to realize the nonlinearity for the double scroll attractor. The outputs of the comparators are directly connected to a Personal Computer via a parallel port.

A bit is generated when the signal passes the subspace V_0 or the subspace V_1 region through the subspace V_T. The bit generator block $\mathcal{S}_3$ is described by

$$\mathcal{S}_3 : \quad \sigma_i(\sigma^0, \sigma^1) = \begin{cases} 0 & \text{if } \sigma^0 = 0, \sigma^1 :_0 \uparrow^1 \\ 1 & \text{if } \sigma^1 = 0, \sigma^0 :_0 \uparrow^1 \end{cases} \tag{5.12}$$

where $\sigma^0 :_0 \uparrow^1$ shows a raising edge (transition from logic value 0 to 1) of σ^0 and $i \in \{0, 1, 2, ...\}$. Figure 5.11 shows the corresponding output bits for an input of the block $\mathcal{S}_3$. This bit generation method basically aims at characterizing the jumps in signal $x(t)$ from one scroll to the other or staying at the same scroll. The threshold value c_1 is set to 0 Volt. c_2 is not determined at this point but is considered as a design parameter to further improve the statistical properties of the bit sequence, as will be further explained in the next section. The signals $y(t)$ and $z(t)$ do not directly affect the bit generation. Therefore, when one compares the proposed RBG with a random number generator using 1D chaotic map, one recognizes that the proposed RBG brings additional difficulties to the opponent in cryptography.

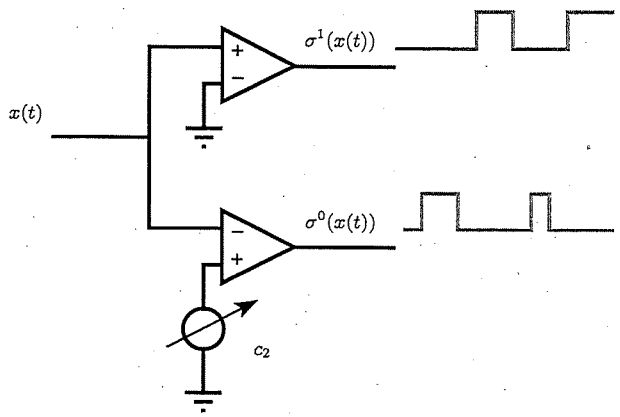


Figure 5.10: Additional circuit to generate binary sequences.

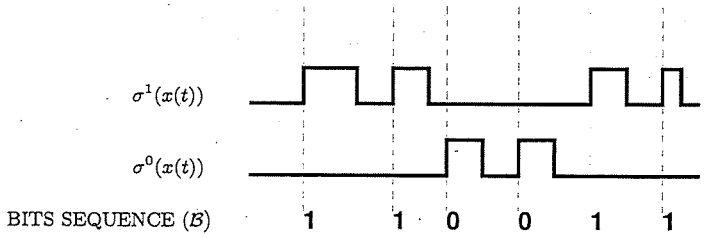


Figure 5.11: A bits sequence from the comparator outputs.

5.3.1.2 De-skewing and setting the threshold

A natural source of random bits may not give unbiased bits as direct output [117, 105]. Many applications, especially in cryptography, rely on sequences of unbiased bits. There are various techniques to extract unbiased bits from a defective generator with unknown bias. These are called *de-skewing techniques* [179]. These techniques also eliminate the correlation in the output of the natural sources of random bits. This problem is also thought as simulating a fair (unbiased) coin using a biased coin. Von Neumann [277] has probably been the first author to state this problem. He proposed a digital post-processing that balances the distribution of bits. Post-processing converts non-overlapping pairs of bits into output bits by converting the bit pair 0 1 into an output 0 and converting 1 0 into an output 1. The pairs 1 1 and 0 0 are discarded [179]. In [66] the authors used the Fast Fourier Transform (FFT) for de-skewing in order to obtain uncorrelated and unbiased numbers from the disk's timing data. Also, cryptographic hash functions are used as another de-skewing technique by taking advantage of their statistical property [179]. In this paper Von Neumann's technique is used because it can easily be integrated into the hardware and it is not decreasing the bit rate too much compared with the other methods. The de-skewing block $\mathcal{S}_4$ which is used in this paper, is described by

$$\mathcal{S}_4: \quad b_i(\sigma_i, \sigma_{i-1}) = \begin{cases} 0 & \text{if } \sigma_i = 0 \bigwedge \sigma_{i-1} = 1 \\ 1 & \text{if } \sigma_i = 1 \bigwedge \sigma_{i-1} = 0. \end{cases} \tag{5.13}$$

In order to obtain an unbiased bit sequence, the bit generator part is combined with a simple subroutine of the de-skewing block $\mathcal{S}_4$. The proposed TRBG has a single output which is denoted by $\mathcal{B} = \{..., b(i-1), b(i), b(i+1), ...\}$ and $b(i) \in \{0, 1\}$. Statistical properties of the TRBG can be adjusted by the threshold c_2 value which serves as a control parameter for the TRBG. Using the fact in information theory that noise has maximum entropy, the threshold value c_2 is chosen such that the measured entropy of the TRBG is maximal.

The measure-theoretic entropy [84] of the proposed true random bit generator with respect to a partition c_2 is given by

$$h^{c_2} = \lim_{n \to \infty} \frac{H_n^{c_2}}{n}$$

where

$$H_n^{c_2} = -\sum_{\mathcal{B}^n} P(\mathcal{B}^n) \ln P(\mathcal{B}^n) \tag{5.14}$$

with $P(\mathcal{B}^n)$ the probability of occurrence for binary subsequence (words $\{..., \{b(i), b(i+1), ..., b(i+n-1)\}, \{b(i+n), ..., b(i+2n-1)\}, ...\}$) of length n [164, 84, 237]. $H_n^{c_2}$ achieves its maximum when the n-word sequences in the output sequence $\mathcal{B}$ of the TRBG are equally distributed.

To find the maximum entropy with respect to the threshold c_2, 1,000,000 bits were generated for 17 candidate thresholds values. Using (5.14), $\frac{H_n^{c_2}}{n}$ was calculated for $n = 3, 4, ..., 14$ Figure 5.12(a) shows values for $n = 3, 4, ..., 10$. (Figure 5.12(b)) shows how the proposed TRBG can come close to the maximum entropy ($\ln 2$) that is achievable for a true random bit generator. Threshold c_2 is therefore set to -1.44 Volt.

5.3.1.3 Statistical tests

A large number of statistical tests and whole test suites have been proposed to assess the statistical properties of random number generators. Truly random sequences are e.g. likely to exhibit the same number of 0's and 1's in the sequence. A finite set of statistical tests may only detect defects in statistical properties. Furthermore, the tests cannot verify the unpredictability of the random sequence which is demanded in cryptographic applications. In this paper unpredictability of the sequence is a consequence of having a chaotic signal as a source.

In this section, the proposed random bit generator is faced with statistical tests. The statistical tests are applied to a bit sequence obtained after the de-skewing. As introduced in the previous section, de-skewing maintains bias and correlation and it also maintains the other statistical properties. A better result of the statistical test can be obtained, if a strong de-skewing technique is used such as a cryptographic hash function [179], but these are more time-consuming.

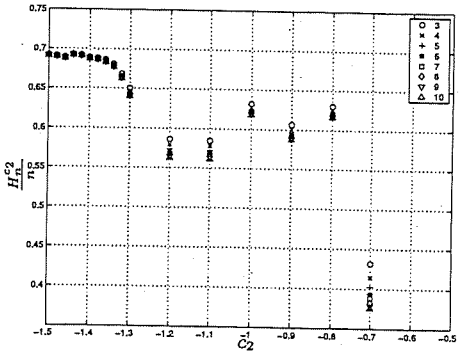
Before we present the test results, the tests that are applied in the next subsection are explained (details of the tests can be found in [179, 137]):

Monobit test : The purpose of this test is to determine whether the number of 0's and 1's in the bitstream are approximately the same.

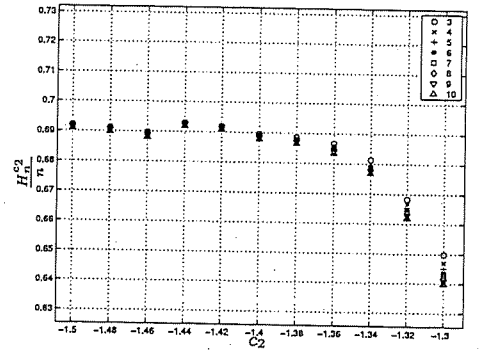
Poker test : The poker test determines whether the sequences of length m appear approximately the same number of times in the bitstream. The bitstream is divided into k ($k = \lfloor \frac{n}{m} \rfloor$) non-overlapping parts each of length m. Let n_i be the number of occurrences of the i^{th} type of sequence of length $m = 1, 2, ..., 2^m$. The statistic is then

$$X_3 = \frac{2^m}{k} \left(\sum_{i=1}^{2^m} {n_i}^2 \right) - k \qquad (5.15)$$

which approximately follows a χ^2 distribution [137, 179] with $2^m - 1$ degrees of freedom.



(a)



(b)

Figure 5.12: (a) $\frac{H_n^{c_2}}{n}$ as a function of the threshold c_2 for $n = 3, ..., 10$; (b) $\frac{H_n^{c_2}}{n}$ for $n = 3, ..., 10$ where $-1.5 \leq c_2 \leq -1.3$.

Runs test : A run of a bitstream is a subsequence consisting of consecutive 0's or consecutive 1's. A run of 0's is called a gap, while a run of 1's is called a block. The runs test determines whether the number of runs of various lengths in the bitstream is as expected for a random sequence. The statistic used is

$$X_4 = \sum_{i=1}^{k} \frac{(B_i - e_i)^2}{e_i} + \sum_{i=1}^{k} \frac{(G_i - e_i)^2}{e_i} \tag{5.16}$$

with $e_i = \frac{n-i+3}{2^{i+2}}$ (the expected number of gaps and blocks of length i) where n is the length of the random sequence and B_i, G_i is the number of blocks and gaps, respectively, of length i in the bitstream for each $i = 1, 2, ..., k$. X_4 approximately follows a χ^2 distribution [137, 179] with $2^k - k$ degrees of freedom. It is noted by Knuth [137] that the runs test is a strong test and is used in [276].

FIPS $140-1$ statistical tests for randomness

Federal Information Processing Standards (FIPS) are issued by the National Institute of Standards and Technology (NIST). FIPS $140-1$ covers security requirements for cryptographic modules and specifies recommended statistical tests for random number generators. Here a single bitstream of length $20,000$ bits as output from the random number generator is subject to each of the tests. If one of the tests fails, the generator fails the test [179].

- *Monobit test :* The number of 1's (N_1) in the bitstream should satisfy $9654 < N_1 < 10346$.
- *Poker test :* The statistic X_3 is computed for $m = 4$. The poker test is passed if $1.03 < X_3 < 57.4$.
- *Runs test :* The number of blocks B_i and gaps G_i of length i in the bitstream are counted for $i = 1, ..., 66$. The runs of length greater than 6 are considered to be of length 6. The runs test is passed if the 12 counts B_i and G_i ($i = 1, ..., 6$) are each within the corresponding interval given in Table 5.1.
- *Long run test :* The long run test is passed if there are no runs of length 34 or more.

The proposed true random bit generator successfully passes all four statistical tests for every run. Table 5.2 provides results from five consecutive runs for the FIPS $140-1$ specified tests.

Length of run	Required interval
1	$2267 - 2733$
2	$1079 - 1421$
3	$502 - 748$
4	$223 - 402$
5	$90 - 223$
6+	$90 - 223$

Table 5.1: Runs test table.

Diehard Test Suite

For further testing, we use the software package called Diehard [172] that has been designed for empirical testing. Diehard includes a battery of about 200 instances of fifteen traditional statistical tests which are birthday spacings, overlapping permutations, ranks of 31×31 and 32×32 matrices, ranks of 6×8 matrices, monkey tests on 20-bit Words, monkey tests OPSO, OQSO, DNA, count the 1's in a stream of bytes, count the 1's in specific bytes, parking lot, minimum distance, random spheres, squeeze, overlapping sums, runs, and craps.

The Diehard test suite has been applied here to 80 million bits (`http://www.esat.kuleuven.ac.be/~mey/Ds2RbG/ds2rbg.bin`) generated by the system proposed in this section. It passes the full suite of Diehard. The complete test results can be further verified at `http://www.esat.kuleuven.ac.be/~mey/Ds2RbG/Ds2RbG.html`

5.3.2 Binary pseudorandom pattern generation on CNN-UM

The aim of this Section is to show how the CNN-UM can be used to generate binary random patterns for applications such as watermarking [308, 9, 309] and analyzing time series of multidimensional binary signals [87]. A binary random pattern contains a set of random bits which are arranged in a grid (here we consider a $N \times M$ grid which is the size of the CNN array). The subsequently generated Black&White pictures have an irregularity in space.

The result of our research and Crounse *et al.* [56] is that the simulation of certain classes of binary Cellular Automata (CA) [285] is the most practical and appropriate way to generate random patterns on CNN-UM. The CA are dynamical systems in which space and time are discrete. In CA, each point in a regular spatial lattice, called cell, can have any finite number of states. The states of the cells in the lattice are updated according to a local rule. In

Run #	Monobit Test	Poker Test	Runs Test			Long Run Test
1	$N_1 = 9943$	$X_3 = 13.9648$	$B_1 = 2614$	$G_1 = 2540$		No long run
			$B_2 = 1201$	$G_2 = 1271$		
			$B_3 = 629$	$G_3 = 624$		
			$B_4 = 313$	$G_4 = 309$		
			$B_5 = 154$	$G_5 = 154$		
			$B_{6+} = 144$	$G_{6+} = 157$		
Passed Test ?	YES	YES	YES		YES	
2	$N_1 = 9813$	$X_3 = 17.0496$	$B_1 = 2683$	$G_1 = 2559$		No long run
			$B_2 = 1224$	$G_2 = 1230$		
			$B_3 = 614$	$G_3 = 633$		
			$B_4 = 280$	$G_4 = 354$		
			$B_5 = 146$	$G_5 = 155$		
			$B_{6+} = 140$	$G_{6+} = 155$		
Passed Test ?	YES	YES	YES		YES	
3	$N_1 = 10011$	$X_3 = 18.1248$	$B_1 = 2536$	$G_1 = 2595$		No long run
			$B_2 = 1269$	$G_2 = 1240$		
			$B_3 = 662$	$G_3 = 610$		
			$B_4 = 291$	$G_4 = 299$		
			$B_5 = 149$	$G_5 = 149$		
			$B_{6+} = 149$	$G_{6+} = 163$		
Passed Test ?	YES	YES	YES		YES	
4	$N_1 = 10052$	$X_3 = 14.9696$	$B_1 = 2513$	$G_1 = 2482$		No long run
			$B_2 = 1215$	$G_2 = 1277$		
			$B_3 = 607$	$G_3 = 634$		
			$B_4 = 343$	$G_4 = 308$		
			$B_5 = 173$	$G_5 = 153$		
			$B_{6+} = 145$	$G_{6+} = 142$		
Passed Test ?	YES	YES	YES		YES	
5	$N_1 = 10068$	$X_3 = 24.4224$	$B_1 = 2471$	$G_1 = 2498$		No long run
			$B_2 = 1318$	$G_2 = 1289$		
			$B_3 = 609$	$G_3 = 653$		
			$B_4 = 299$	$G_4 = 284$		
			$B_5 = 162$	$G_5 = 155$		
			$B_{6+} = 161$	$G_{6+} = 141$		
Passed Test ?	YES	YES	YES		YES	

Table 5.2: Results of FIPS−140 − 1 specified tests.

other words, at a given time the state of a cell depends on its own previous states and on the states of its neighbouring cells. If the next-state transition function, expressed algebraically, is linear, then the CA is called additive CA, otherwise, it is called non-additive CA. Note that different cells may use a different or identical local rule, making the CA hybrid or uniform, respectively. CA may possess a fixed boundary condition or a periodic boundary condition. For more information about CA the reader is referred to [285]. In fact, CA and CNN share a common property; namely, their dynamic behavior depend only on their spatial local interactions. The main difference is that for CA the state variables are binary valued function for the dynamics is a logic function, whereas for CNN, the state variables are real-valued variables and the function for the dynamics is a real valued function.

Here, only binary one- and two-dimensional CAs with two neighbor cells (left and right) and five-neighbor square (left, right, up, down and itself) will be respectively considered. These correspond to the sphere of influence Eq. (2.11) $\mathcal{S}_i(1)$ and $\mathcal{S}_{i,j}(1)$ when the CNN is considered, respectively. Furthermore, it has been demonstrated that the CNN-UM is capable of implementing an arbitrary CA from this class [55]. In order to do that logic operational capability of CNN-UM or by cascading the outputs of threshold logic units [55] which are used to implement linearly separable boolean functions on the analog array of CNN-UM can be employed to implement any arbitrary Boolean function.

We proceed with describing several different CA-based Pseudorandom Pattern Generator (PPG) algorithms. These are classified into two groups: one-dimensional CA (1-D CA) rules and two-dimensional CA (2-D CA) rules. 1-D CA rules have been extensively studied in [115] and Rule 30 and 45 (see details in the following Sections) have been successfully used for image and video watermarking in [308] and [9], respectively. Here we review these two rules and introduce additional rules: Rule 90 and 150 on hybrid CA [115]. In [56] a PPG algorithm for CNN-UM was proposed based on an uniform 2-D CA rule, and it was shown that CNN-UM can produce a two dimensional pseudorandom pattern by means of CA.

5.3.2.1 PPGs from 1-D rules

A simple and straightforward way to use 1-D CA rules on the CNN-UM is to think of rows and columns of a CNN array as 1-D CAs. So, 1D-CA rules can be applied to all rows

$$W_{i,j}(n) = f_{\text{row}}(W_{i,j-1}(n-1), W_{i,j}(n-1), W_{i,j+1}(n-1)) \tag{5.17}$$

and columns

$$W_{i,j}(n+1) = f_{\text{col}}(W_{i-1,j}(n), W_{i,j}(n), W_{i+1,j}(n)) \tag{5.18}$$

where $W_{i,j} \in \{0,1\}$, $i \in \{1,2,\cdots,N\}$ and $j \in \{1,2,\cdots,M\}$. Rule 30 and Rule 45 [285, 36] for uniform non-additive CA and Rule 90 and Rule 150 for hybrid additive CA [36] are candidate rules to generate the pseudorandom pattern.

Rule 30 and Rule 45 on uniform CA

The dynamics of Rule 30 and Rule 45 are

$$a_i(n+1) = f_{30} = a_{i-1}(n) \oplus (a_i(n) \vee a_{i+1}(n)) \tag{5.19}$$

and

$$a_i(n+1) = f_{45} = a_{i-1}(n) \oplus (a_i(n) \vee \overline{a}_{i+1}(n)), \tag{5.20}$$

with $\overline{a}$, $\oplus$ and $\vee$ denote logic "not", "exclusive-or" and "or" operations, respectively. Pseudorandom patterns are generated when the same rules are consecutively applied to the rows and columns of the CNN-UM. The PPG can be formulated as follows

$$W_{i,j}(n) = f_{\text{row}} = f_{30}$$

$$W_{i,j}(n+1) = f_{\text{col}} = f_{30}$$

which is called H30V30. H45V45 can be similarly constructed. It is also possible to use a different rule on the rows and columns, called H30V45 and H45V30.

Rule 90 and Rule 150 on hybrid CA

The dynamics of Rule 90 and 150 are

$$a_i(n+1) = a_{i-1}(n) \oplus a_{i+1}(n) \tag{5.21}$$

and

$$a_i(n+1) = a_{i-1}(n) \oplus a_i(n) \oplus a_{i+1}(n). \tag{5.22}$$

In [115], it has been experimentally shown that the hybrid constructions are necessary to achieve a CA with maximal cycle length. The hybrid CA are configurations chosen with a rule vector $T = [d_1 \;\; d_2 \;\; \cdots d_N]$ where

$$d_i = \begin{cases} 0 & \text{if cell i uses rule 90} \\ 1 & \text{if cell i uses rule 150.} \end{cases}$$

For example, consider a hybrid CA of length 10 with the rule vector $T =$ [0 1 0 0 0 0 1 0 0 0] where cells 2 and 7 use Rule 150 and the rest use Rule

90. The rule vectors for linear hybrid CA with fixed boundary condition of up to 500 cells have been given in [26] to achieve maximum cycle length. Again these rules can apply to rows and columns of CNN arrays to produce a random pattern like in the previous Section. This algorithm is called HV90150.

5.3.2.2 PPGs from 2-D rules

Although PPG by 1-D CA has been intensively studied, 2-D CA for PPG has not been sufficiently analyzed in literature. The major work on the 2-D CA for pseudorandom pattern generators [36] shows that 2-D CAs are better than 1-D CA of the same size and also Linear Feedback Shift Register (LFSR) in terms of the quality of randomness. Recently the same problem has been addressed in [266] to find the rule for PPG on hybrid 2-D CA using genetic algorithms.

A PPG algorithm on the CNN-UM based on a uniform non-additive CA was first proposed by Crounse *et al.* in 1996 [56]. They have introduced the following dynamics

$$\begin{aligned} W_{i,j}(n+1) = {} & (W_{i+1,j}(n) \vee W_{i,j+1}(n)) \oplus W_{i-1,j}(n) \\ & \oplus W_{i,j-1}(n) \oplus W_{i,j}(n) \end{aligned} \tag{5.23}$$

named PNP2D.

Although it is possible to achieve the maximum cycle length by using a 2-D hybrid CA [36], there is no set of rules which has been reported for the pseudorandom pattern generation on a large grid size of a 2-D CA (e.g. 64×64). Therefore, here we do not explore 2-D hybrid CAs.

5.3.2.3 Experimental results on the CNN-UM chip

The proposed PPG algorithms (H30V30, H30V45, H45V45, PNP2D and HV-90150) have been implemented in AMC language and tested on the 64 × 64 CNN-UM chip (ACE4k) using a professional CCPS system [320]. The implementation of CA rules on the CNN-UM involves two groups of operations: template based (analog) and logic based operations between the Local Logic Memories (LLMs). Template operations are used to shift the contents of the CNN array into required directions with periodic (2.15-2.18) or fixed boundary (2.23-2.26) with $E_1 = E_2 = E_3 = E_4 = 0$ condition. The results obtained (W_n) are shown in Figure 5.14 for iteration steps $n = 5, 10, 15, 20, 25, 30$ and an initial condition in Figure 5.13. The Black&White pixels of the obtained images uniformly distributed after 30 iteration steps (the last row of Figure 5.14).

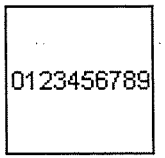


Figure 5.13: An initial image for the PPG algorithms.

The template role is to shift the contents of the CNN array to the right direction $(W(i,j-1))$

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -3 & 0 \\ 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 & 0 & 0 \\ 3 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, z = 0.$$

Shifting to other directions is implemented by moving the non-zero term of the B template in clockwise direction. Furthermore, the following template is used to shift and bitwise inverse the content of the CNN array to the right $(\overline{W}_{i,j-1})$

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -3 & 0 \\ 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 & 0 & 0 \\ -3 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, z = 0.$$

Evolution in time of the algorithms shows that the H30V30 algorithm provides a good result in a short iteration time, which is a similar result as in [36]. In order to implement R90150 on the CNN-UM, configuration vectors for columns $T_c \in \mathbb{B}^{1\times N}$ and rows $T_r \in \mathbb{B}^{1\times M}$ are listed as entries of binary matrices $T_{\text{col}} = [T_{c_1}^T\ T_{c_2}^T \cdots T_{c_M}^T]$ and $T_{\text{row}} = [T_{r_1}^T\ T_{r_2}^T \cdots T_{r_N}^T]^T$ where all the columns $(T_{c_i} = T_c)$ and rows $(T_{r_i} = T_r)$ are the same as the configuration vectors, respectively. The outcomes of Rule 90 and Rule 150 are obtained on LLMs; the final result can be obtained by a simple logic operation between LLMs, T_{col} and T_{row}. In fact, the matrices T_{col} and T_{row} can be used as the fixed state maps in order to implement the algorithm on the CNN-UM. The fixed state option allows to perform a template operation on specified cells in the CNN array of the CNN-UM by defining the fixed state map. Although HV90150 gives a longer cycle length than other 1-D CA rules, it needs a longer iteration time to obtain a good randomness in space.

A lot of runs of the algorithms on the ACE4k chip gave the same results, confirmed by a CNN simulation. The number of iterations is experimentally tested to obtain uniformly distributed Black&White pixels. The minimum number (N_{m}) of about 30 iteration steps is needed to obtain a good random

pattern. However, as we have noted, initial conditions are important to obtain a random pattern (or number) for deterministic algorithms like the proposed algorithms.

The timing of the proposed algorithms is given by the following equations:

$$\begin{aligned} \tau_{\text{H30V30}} &= (4\tau_{logic} + 4\tau_{temp} + 5\tau_{copy}) \\ \tau_{\text{H30V45}} &= (4\tau_{logic} + 4\tau_{temp} + 5\tau_{copy}) \\ \tau_{\text{H45V45}} &= (4\tau_{logic} + 4\tau_{temp} + 5\tau_{copy}) \\ \tau_{\text{PNP2D}} &= (4\tau_{logic} + 4\tau_{temp} + 7\tau_{copy}) \\ \tau_{\text{R90150}} &= (12\tau_{logic} + 4\tau_{temp} + 15\tau_{copy}) \end{aligned}$$

where τ_{logic}, τ_{temp} and τ_{copy} are the required time to perform a logic, template and copy operation on the CCPS system. On the CCPS system, $\tau_{\text{H30V30}} = \tau_{\text{H30V45}} = \tau_{\text{H45V45}} = 237.2\mu s$, $\tau_{\text{PNP2D}} = 307.2\mu s$ and $\tau_{\text{R90150}} = 617.6\mu s$.

5.3.2.4 A general comparison

The proposed PPG algorithms are designed to optimally run on the CNN-UM. Therefore, a direct comparison among the algorithms on different hardware platforms with completely different architectures is not fair. However, for a general comparison between the CNN-UM approach and the traditional ones, let us consider to implement the same algorithm on a traditional processing machine, with a core to process the data and a bus to exchange data to/from memory. In this way we can make some estimations of hypothetical timings on classical systems. Basically, the way of indexing the memory and processor speed are the features to evaluate and compare our approach with the other technologies; the basic operations of our approach are shift and logic operation. These ones require a high number of memory accesses in traditional systems, because they have no internal register capable of storing this amount of information [309]. For example to shift a chipsize image in whatever direction the memory controller reads and writes 8192 ($64 \times 64 \times 2$) times from/to the memory. This means that any operation access-to-memory number equals the double chipsize. A complete analysis of the access time latency in a memory chip is beyond the topic of this paper, however, the sum of the t_{RP} (Time Row Precharge), t_{RCD} (Time Ras to Cas Delay; Ras: Row Access Strobe; Cas: Column Access Strobe) and t_{CAC} (Column Access Time) is about 15 cycles (in fastest memory chip (see `http://www.geilusa.com`)) of the Front Side Bus. Therefore, in a novel DDR400MHz chip $37.5ns$ are spent to retrieve data from the memory. This means that to calculate the time spent to move 8192 bytes from/to memory this simple formula could be applied:

$$37.5\ ns \times 8192/4 = 76.8\mu s.$$

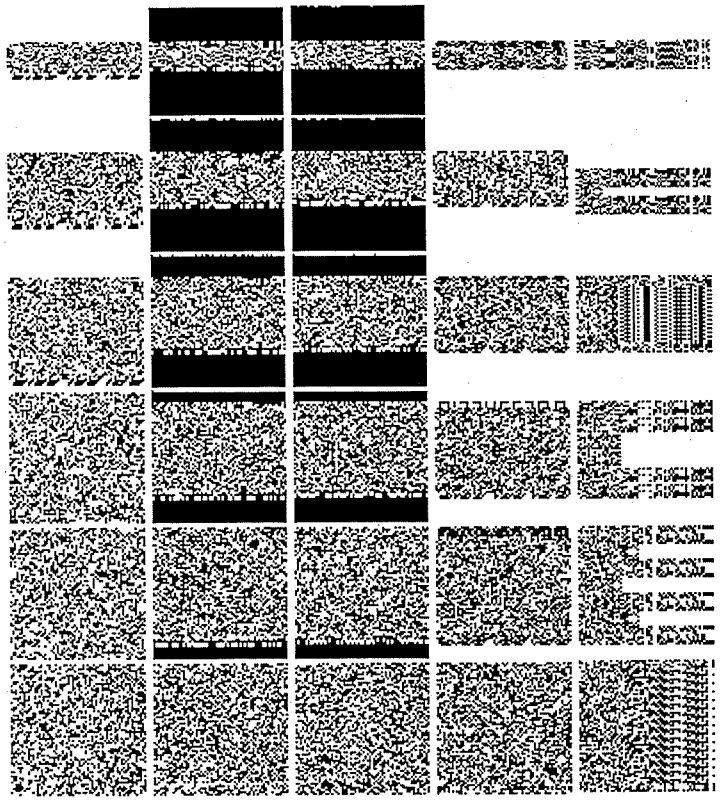


Figure 5.14: Evolution of H30V30 (1st column), H30V45 (2nd column), H45V45 (3rd column), PNP2D (4th column) and R90150 (5th column) algorithms for the iteration steps $n = 5, 10, 15, 20, 25, 30$ (top to bottom). The Black&White pixels of the obtained images uniformly distributed after 30 iteration steps. These images can be used as a watermark.

The division by 4 is required because a DDR memory reads/writes 4 byte (32bits) per each reads/writes. Taking into consideration the H45V45 algorithm (but the same considerations also hold for the other algorithms), it is composed by the following operations: 4 logic operations, 5 template operations, 5 copy operations. These are iterated in a cycle composed by 64 iteration steps. Summarizing the whole time period equals:

$$76.8\mu s \times (4 + 5 + 5) \times 64 = 68.8ms.$$

This timing is about twenty times our timing; it does not satisfy the real time requirement ($< 25ms$). Moreover, it has to be outlined that this is the time spent only to perform the memory access without considering the processing time, i.e. assuming an infinite speed of the processor. From these considerations the analog distributed approach based on a CNN-UM system is very promising, since it allows the processing of a whole chip size image, minimizing the memory access number.

5.4 Image/Video authentication on CNN-UM

The digital information revolution has brought many advantages and new issues in this century. With the ease of editing and perfect reproduction, the proof of ownership and prevention of unauthorized manipulation of digital images and video materials has become an important issue. The objective of authentication is to enable detection of originality of the digital media and preventing forgery. The authentication algorithms have to be processed just after an image is captured. Hence, there is a need to develope a camera which can be also operate with the authentication algorithm inside, such as trustworthy digital cameras [85]. In this Section we show that the authentication algorithm can be implemented on the presently available CNN-UM.

The digital watermarking [284, 157, 160] and cryptographic data authentication [179, 201] provide an adequate solution to the authentication of digital images and videos. We address the watermark-based solution due to some of the reported advantages of watermarking-based authentication algorithms such that [18];

- Cryptographic data authentication algorithms are constrained to the use of a particular image format, making it impossible to use different formats.
- It is impossible to distinguish between malicious and innocuous modification.

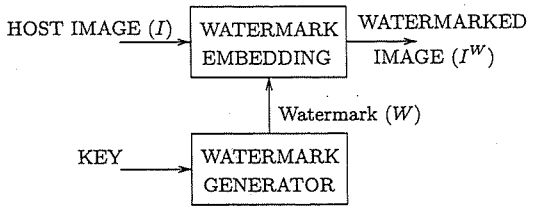


Figure 5.15: Watermark embedding.

- It is also not possible to locate modifications made to an image.

Nevertheless, based on cryptographic data, authentication algorithms can be found in [308, 60, 295, 59] for the CNN-UM.

In the literature, many types of watermarking have been developed for a variety of applications. Watermarks may be visible or invisible: a visible mark is easily detected by observation, while an invisible mark is designed to be transparent to the observer and detected only by using signal processing techniques. In the class of invisible watermarks, one may further distinguish categories of robust and fragile techniques [284, 157, 160]. A robust mark is designed to resist attacks that attempt to remove or destroy the mark, meanwhile a fragile mark is designed to detect slight changes to the watermarked image with high probability. An innovative application of fragile watermarking is the steganography, in which the watermarking hides a message through a transparent mark. In this case the image is used as a medium for transfering a secret message inside.

In this Section, a fragile watermark method is proposed for image and video authentication problems on the CNN-UM. This method is the most commonly used technique in the literature. In particular, the CNN-UM will be used in the watermark embedding part, which has two crucial blocks: the watermark generator and the embedding block (Figure 5.15). In this method, the watermark is a pseudorandom pattern that is generated based on a key (which is an initial image). Here, we use watermark generators based on a one-dimensional and two-dimensional cellular automaton as proposed in Section 5.3.2. The way to embed the watermark into an image is to add the watermark to the luminance values of its pixels.

5.4.1 Fragile watermarking

A fragile watermark is a mark that is readily altered or destroyed when the host image is modified. Fragile watermarking is not suited for enforcing copyright ownership of digital images. The basic conceptual structure for embedding and detecting a fragile mark is shown in Figure 5.16. It is similar for any watermarking system. An owner embeds the mark into an original image (host image) (Figure 5.15). The marking key is used to generate the watermark and is typically assigned as identifier to the owner or to the image. The marked image is perceptually identical to the original image under the normal observation, and may be transmitted or distributed. The receiving user has a "detector" (Figure 5.16) to evaluate the authenticity of the received image. The detection process also requires knowledge of side information. The side information may be the marking key, watermark, original image, or other information. The detector is usually based on statistical detection theory: after its application, the image is determined to be authentic or not. It would be desirable for the detector to determine where the image has been modified if the image is not authentic.

There are two basic modalities for fragile watermarking: spatial domain [278, 271, 318, 282, 283, 178] and transform domain such as the Discrete Cosine Transform (DCT) domain, Discrete Fourier Transform (DFT) domain and Discrete Wavelet Transform (DWT) domain [289, 293]. Spatial domain techniques provide simple and effective schemes for embedding an invisible watermark into the original image. The simple method of spatial domain watermarking technique is the Least Significant Bit (LSB) modification method [178] which embeds the mark in the LSB plane for perceptual transparency. Its main significant disadvantage is that the whole image can be modified while keeping the LSB intact. Better and more secure schemes can be constructed using pseudorandom noise patterns. Van Schyndel [271] describes a straightforward way to add a watermark to an image in the spatial domain by adding a pseudorandom noise pattern, which will be the watermark, to the luminance values of the image pixels. Wolgan and Delp [282] extended the work of Van Schyndel [271] to improve robustness and localization in their Variable-Watermark Two-Dimensional (VW2D) technique. We refer to [284, 157, 160] for further information about watermarking and fragile watermarking.

5.4.2 Fragile watermarking on the CNN-UM for image authentication

Here, instead of dealing with non-CNN friendly operations such as FFT, DCT and DFT [260] and the computational complexity of the transformations, a

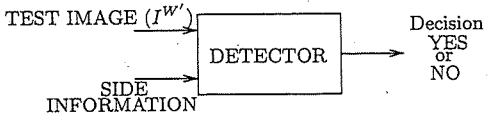


Figure 5.16: Watermark detection.

spatial domain watermarking technique is proposed for the CNN-UM, which is based on [271] and [283]. The watermark is generated based on a key (marking key). The only constraints are that the energy in the pattern is more or less uniformly distributed and that the pattern is not correlated with the host image content. To create the watermarked image (I^W), the pseudorandom pattern (watermark (W)) is multiplied by a small gain factor k and added to the host image (I),

$$I^W_{i,j} = I_{i,j} + kW_{i,j}. \tag{5.24}$$

To detect a watermark in a possibly watermarked image ($I^{W'}$), the correlation between image $I^{W'}$ and watermark is calculated. To give a correct decision about authentication of image, the correlation value with correct key must be higher than for wrong keys. Nevertheless, robustness of the watermark is not a key issue in fragile watermarking: the robustness can be improved significantly by increasing the gain factor. This technique only offers to detect authentication of images. In some applications, it is also important to locate any alteration made on the image. Therefore, the correlation value is not calculated on the whole image but on block-by-block basis between the image and the watermark. This is basically a VW2D type of watermarking which as given in [283], offers to detect and locate a change made on an image.

The watermark embedding process is easily implemented on the CNN-UM. The main advantage for using the CNN-UM is the fast generation of a watermark and the addition to the original image. Hence it saves the time for further or/and image processing algorithm(s). The CNN-UM with a fragile watermarking algorithm can be thought of as the trustworthy digital camera as proposed by Friedman in 1993 [85]. Trustworthy digital camera is the first proposed system to protect image authentication on a digital camera. It is simply a digital camera and additionally it has a microprocessor to compute a cryptographic hash value [179, 201] of a captured image for authentication. The CNN-UM with optical sensing can be further exploited over trustworthy video camera for its image processing capacity. Here, watermark detection is not considered because of its dependence on the particular computation technique which questions the authentication of a received image. Therefore, this method should be compatible with any receiver which is able to compute correlation between two images.

5.4.3 Fragile watermarking on the CNN-UM for video authentication

Video watermarking is considered as marking the consecutive image sequences using the same technique as in the previous Section. It consists of adding an original frame of I to pseudorandom pattern (W) multiplied by a small gain factor k

$$I^W_{l,i,j} = I_{l,i,j} + kW_{l,i,j} \tag{5.25}$$

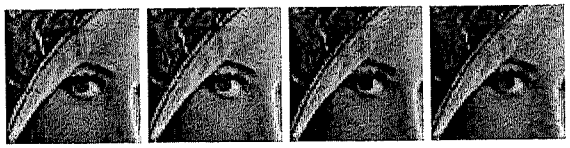
where $l \in \{1, 2, \cdots, L\}$ is the frame number in the video. Hence the video watermark embedding operates on the spatial axis as image watermarking and additionally it operates on the temporal axis due to the number of frames in the video sequence. Due to the complexity of the detection scheme and the need to save the watermarks for each frame, a single watermark $W_{l,i,j} = W_{i,j} \ \forall l$ might be used for every video frame as suggested in [168] for DVD video watermarking, which is called as frame-independent watermark. However, the main drawback of this approach, reported in [18], is that a frame-independent watermark can be found by a comparative analysis of all image sequence frames and then it could be easily added to fake frames. Therefore, a feasible system must be capable of generating and embeding a new watermark into each frame satisfying the real time requirement. The fragile watermarking algorithm which has been proposed above for video authentication is computationally simple and straightforward to be solved on the CNN-UM. Like in image authentication, it basically contains two blocks which are the watermark generator and the watermark embedding. Watermark embedding consists simply of adding each watermark, scaled by k, into the corresponding host frame until the whole video sequence is marked. The computational expense for watermark embedding process on the CNN-UM is if compared with the watermark generator process. Our experimental observations have shown that watermark embedding takes $46\mu s$ on the ACE4k chip with a professional CCPS system [320].

5.4.4 Experimental results on the CNN-UM chip

The proposed fragile watermarking algorithm has been implemented in the AMC language and tested on a ACE4k using a professional CCPS system [320]. A chip size part of the well-know *lena* image (Figure 5.17(a)) is marked with the watermark obtained by the H30V30 algorithm.

The following code implements the watermarking embedding algorithms of video frames on the CNN-UM:

```
Line  1:  n = 0;
Line  2:  W(0) = KEY;
```



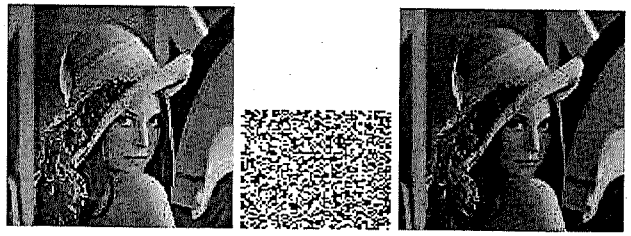
(a) (b) (c) (d)

Figure 5.17: (a) A chip size part of *lena* image (b)-(d) Watermarked images with $k \in \{5, 10, 15\}$ gain factors, respectively.

```
Line  3:  while (n < N_m) {
Line  4:  W(n+1) = watermark_generator(W(n));
Line  5:  n = n + 1;}
Line  6:  i = 0;
Line  7:  while (i < L) {
Line  8:  W(N_m+i) = watermark_generator(W(N_m+i-1));
Line  9:  watermarked_frame(i) = k * W(N_m+i) + frame(i);
Line  10:  i = i + 1;}
```

The watermarking embedding algorithm is consists of two *while cycles*: the first one composes the watermark generator (or PPG), the second one embeds the watermark in the frame. It has to be stressed that each frame is marked by a different watermark: in `Line 8` the new watermark is generated and in `Line 9` it is added to the frame. However, using the same watermark in `Line 9` (`w(`N_m`)` instead of `w(`N_m`+i)`), the algorithm becomes frame-independent. Because of the high computational power of the CNN-UM, it is also possible to mark the frames with a watermark which is generated by different keys. This algorithm has been totally implemented on the ACE4k board. The main part is performed by the CNN chip, and only one operation is executed on the DSP: the addition (`Line 9`). Figures 5.17(b)-(d) show the results after watermarking Figure 5.17(a) with $k = 5/256, 10/256, 15/256$, respectively.

A time evaluation shows that this operation is performed in about 46μs. Using four tiling operations that permit to allocate images with maximum dimension of 256×128 pixels on the local board memory, dividing it into several sub-images 64×64, processed in a serial way. The 128×128 *lena* image in Figure 5.18(a) has been watermarked using the same watermark (Figure 5.18(b)) in each tiling operation. Hence, the image is marked on a block-by-block basis. The watermarked image is given in Figure 5.18(c). The total timing for



(a) (b) (c)

Figure 5.18: (a) A host image (*lena* image); (b) a watermark from H30V30 algorithm; (c) a watermarked image for the gain factor $k = 2$.

watermarking a 128×128 *lena* image is $184\mu s$.

Each frame of the well-known "table tennis" video has been watermarked using a H30V30 watermark generator. Using four tiling operations, each 128×128 video frame has been watermarked. Watermarking of resized "table tennis" video can be seen in `http://www.esat.kuleuven.ac.be/~mey/dictam/project.html`. Like the watermarking "lena" image (Figure 5.18(c)), we were able to embed a watermark into the still frame without any visible artifact.

5.5 CNN template tuning

Presently, analogic CNN Visual Microprocessors are supporting Tera operations per second equivalent computing speed, and rates of more than 10,000 frames per second rate have been tested [214]. This performance can make CNN chips very suitable for a wide range of image processing tasks. Several design [319] and learning methods [153, 27, 101] were developed to generate CNN templates for execution of different tasks. These methods were based on ideal CNN structures and the resulting templates could perform a wide range of operations when executed on simulators that are implemented on digital computers. When these templates were tested on the implemented CNN-UMs, many of these templates were found to work incorrectly in their original form [83]. The causes of that are mainly erroneous chip behaviors. Differences between ideal structures and real chips prevent the chips to react in the same way as the simulators and cause sometimes serious differences between simulator results

and chip results. Consequently new *chip-independent* robust template design methods were developed [191, 180, 136] for the purpose of generating templates that are more tolerant against inherent parameter deviations and noise in VLSI analog implementations without taking into account specific characteristics of an individual chip.

5.5.1 Erroneous chip behavior and template robustness

The erroneous behavior observed in VLSI implementations of CNN-UM may be caused by a combination of reasons that includes manufacturing process variations and environmental effects (such as temperature and noise). Although adaptive techniques have been employed in CNN-UM chip implementations to ensure accurate external control and system robustness against parameter variations, analog VLSI implementations can only guarantee a rough accuracy (5-10%) in relation to ideal parameter values and, also, template parameters have a discrete range of implementable values (about 1 sign + 7 data bits for the actual chips (see Table 2.1)). In addition to errors caused by manufacturing process variations and environmental effects, some templates developed for use in ideal CNN structures, like the AVERTRSH (average with binary output) [260], can even produce different erroneous results for runs with the same input and initial conditions for a given chip (see Figure 5.19). These so far unavoidable and undesirable features make CNN-UM chips loose reliability for certain template operations. Figure 5.19 clearly shows that the source of errors is not only due to manufacturing imperfections but also susceptibility to post-manufacturing interference. The reasons then could be noise in the components or temperature variation. However, another reason for the inconsistent behavior in Figure 5.19 may also be imperfect or noisy loading of the input and initial state from off-chip to on-chip memory prior to an operation, which may also contribute to the overall undesirable chip behavior.

Methods that consider the design of robust templates have been extensively developed in order to avoid or minimize the affects of these obstacles for correct operation of analog VLSI implementations of CNN-UMs [106]. There are several definitions of robustness [191, 106] that in general define a measure for susceptibility of CNN templates to modifications in their values while still producing the correct output. Basically, robust templates are expected to have their values in the middle of the correct operation interval so that tiny variations on this values are still within the interval. Methods for designing or learning of robust templates generate templates operations with different degrees of robustness. The most robust operations can often be employed in chip implementations with minimal disturbance of the correct output and therefore qualifying these methods as sufficiently good for these template operations. Despite of these advantages, methods for generation of robust templates (or chip-independent methods) can not avoid erroneous behavior on CNN-UM analog VLSI implementations for other less robust template operations. Nonetheless, making use of information (measurements) of a specific chip, one can achieve better results for this chip when these templates are tuned (or calibrated) [291, 292].



(a) (b) (c) (d)

Figure 5.19: For the same input image (a) initial state, and template values, the same chip (an ACE4k) can produce different outputs (b,c, and d). These results were obtained for subsequent runs, what make temperature and noise interferences mostly stable and therefore the different outputs are assumed to be due to imperfect loading from off-chip to on-chip memories.

5.5.2 Experiments

In [291, 292], Xavier de Souza *et al.* uses measurements of actual CNN-UM chips as part of the cost function for a global optimization method to find an optimal template given an initial approximation. Further improvements are achieved by generating chip-specific robust templates by doing a search for the best template among the optimal ones. The tuned templates are therefore customized versions that are expected to be much less sensitive to imperfections on the operation of CNN-UM chips. The experiments were performed using a professional CCPS system [320] in connection with the Matlab environment. The main features of the global optimization algorithm were written for Matlab and were triggered by a AMC program running on the given CNN-UM chip. The input and initial state for each optimized template operation were in general random binary or gray-scale images. The respective desired output was obtained from simulators of ideal CNN-UM using robust templates available in [260].

Among the experiments [292], the cloning template of AVERTRSH operation is chosen here to demonstrate the results of the method on the the ACE4k

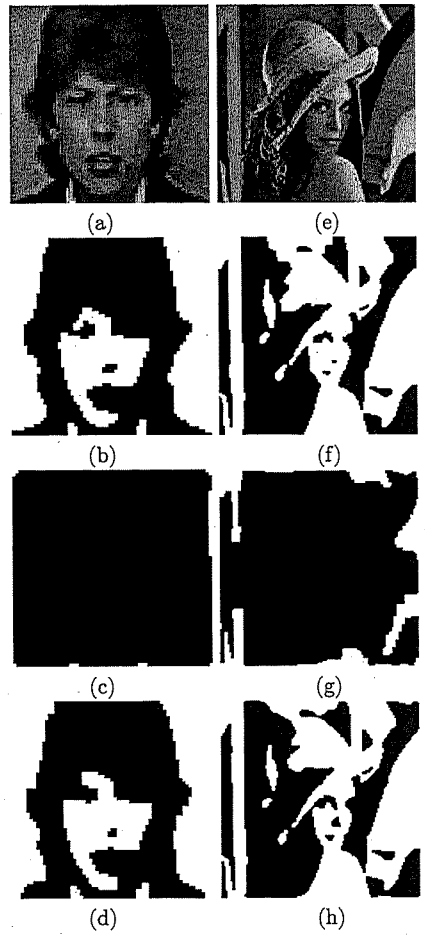


Figure 5.20: Results for average with binary output template operation on the (a-d) ACE4k and (e-h) ACE16k chips. (a,e) Input and Initial state images. (b,f) Desired outputs from the simulator. The chips outputs (c,g) for the templates from the library and (d,h) for the chip-specific optimized templates.

and ACE16k chips. The cloning template values from the template library for AVERTRSH operation are

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, z = 0.$$

After the chip-specific optimization, the resulting cloning templates values for ACE4k and ACE16k are

$$A = \begin{bmatrix} -0.5005 & 1.8088 & -0.1728 \\ 1.7479 & 2.9586 & 1.6937 \\ -0.2007 & 1.8192 & -0.4994 \end{bmatrix}, B = \begin{bmatrix} 0.3354 & 0.7805 & -0.1653 \\ 0.6249 & 0.8771 & 0.7702 \\ -0.2512 & 0.5741 & 0.4052 \end{bmatrix},$$

$z = -2.5131$ and

$$A = \begin{bmatrix} 0.3754 & 2.6839 & 0.3195 \\ 2.9885 & 5.3824 & 2.9885 \\ 0.3195 & 2.6839 & 0.3754 \end{bmatrix}, B = \begin{bmatrix} 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 4.1252 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 \end{bmatrix},$$

$z = -3.1911$, respectively. Figure 5.20 shows the results for these templates on the ACE4k and ACE16k.

5.6 Conclusions

In this Chapter, we faced the question of how the results of the previous Chapters can be used in different applications. We first discussed chaos communication. In the beginning of the 90s, the idea to use chaotic signals to transmit information has appeared after it has been realized that two chaotic systems can be synchronized. Since then there has been significant intereset in exploiting chaotic dynamics in communications. We have provided the reader with an overview on schemes that have exploited chaos for the modulation and demodulation blocks in communication systems. Then we discussed the role of synchronization schemes from Chapter 4 for these blocks. Our discussions have taken into account the recent results of the studies on chaos communications. Hence new synchronization techniques can preserve synchronization against the reported affects in those studies, such as parameter mismatch and propagation delay. In order to take advantages of the developed synchronization schemes in Chapter 4, we have also provided a variety of chaos oscillators and their circuit implementation in Chapter 3. Our experimental results on 2-scroll and 5-scroll attractors have confirmed the theoretical studies of synchronization schemes and also practical use of the proposed chaos oscillators.

A new application for chaos has been found in optimization. A local search mechanism of a simple gradient network for solving continuous nonlinear optimization has been improved by chaotic dynamics. For this purpose, a multi-scroll hyperchaotic attractor from a 1-D CNN consisting of coupled Chua's circuits has been used to explore the search space. The same idea has been also used in CLMs methodology and a wave computing based optimization method has been developed. Simulation results on a number of test functions indicate a good performance of this method with low variance on many different runs.

Although RNGs have been presented as a possible application of chaotic signals for several years, the use of a continuous time signal from a chaotic circuit was never shown with fully empirical tests results. In this Chapter, we presented a random bit generator which uses a double scroll attractor with a simple circuit model. The proposed TRBG is subjected to statistical tests using the well-known tests FIPS-140-1 and Diehard in cryptography. The proposed TRBG successfully passes all four statistical tests of the FIPS-140-1 test suite for all runs. A strong and more complete test suite Diehard has been applied to 80 million bits. The system successfully passes the full suite of Diehard test. Another essential result of this Chapter is that the proposed TRBG implementation can be fully integrated. Therefore, it may become a standard part in hardware.

Furthermore, we have investigated and implemented a watermark-based image/video authentication scheme on a CNN-UM chip. For this purpose, a watermarking scheme has been presented. The proposed method can be included among the fragile watermarking techniques and the CNN-UM is used to generate a watermark and to add a host image in this technique. Possible CA rules for pseudorandom pattern generators are implemented as a watermark generator on a CNN-UM chip.

Finally, we have faced on a side effect of VLSI implementations of CNN-UM chips, related to erroneous behavior. Instead of seeking the reasons of erroneous behavior on the highly complex CNN-UMs chips, we directly approached the problem by generating tuned templates. Results were presented for the binary and gray-scale input cases.

Chapter 6

General conclusions and future work

In this thesis we focused on various intriguing phenomena of nonlinear dynamical system that can be employed towards engineering applications. An important class of nonlinear systems studied in this thesis are Lur'e systems. After a short introduction of our motivations to study nonlinear dynamical systems, we have reviewed cellular neural/nonlinear networks. We put emphasis on recent technological developments of CNN-UMs and new research aspects in CNNs. Then we have given an extensive overview of multi-scroll chaotic attractors with their circuit implementations. Multi-scroll hyperchaotic attractors have been obtained by taking multi-scroll chaotic attractors as cells in an 1-D CNN. These multi-scroll chaotic and hyperchaotic attractors are representable in Lur'e form. As a result all synchronization methods that have been developed for Lur'e systems are applicable to it. For autonomous synchronization schemes, the link with absolute stability theory has been emphasized and robust synchronization has been discussed. For non-autonomous schemes, nonlinear H_∞ synchronization has been explained. Criteria for schemes with time delay have been given which can be either delay-independent or delay-dependent. Sufficient conditions for global synchronization have been expressed in terms of matrix inequalities and have been illustrated on a number of examples. We have discussed the role of synchronization schemes for chaos communication. We showed that the introduced synchronization schemes can overcome the effects that cause loss of synchronization in chaos communication. For example robust synchronization and time-delay synchronization schemes are designed to cope with the effects of influence of parameter mismatch and propagation

delay on the communication channel, respectively. In order to overcome the shortcomings of gradient networks and coupled local minimizers for solving continuous nonlinear optimization, chaotic annealing and coupled chaotic annealing methods have been discussed. These methods use chaotic dynamics to explore the search space for the global minimum. These methods were interpreted within the CNN framework. Furthermore, the coupled chaotic annealing algorithm was represented as an active wave computing algorithm which is controlled by state synchronization. A most encouraging application of chaotic signals in engineering is its use as a source for random number generators. We presented a random bit generator based on a double scroll attractor in Chapter 3. The proposed TRBG is subjected to statistical tests using the well-known tests suites FIPS-140-1 and Diehard in cryptography. The proposed TRBG successfully passes all four statistical tests of the FIPS-140-1 test suite for all runs. A strong and more complete test suite Diehard has been applied to 80 million bits. The system successfully passes the full suite of Diehard test. As an application of CNN-UM, a watermark-based image/video authentication scheme has been investigated and implemented on a CNN-UM chip. The proposed method can be included among the fragile watermarking techniques. The CNN-UM was used to generate a watermark and to add a host image in this technique. Possible CA rules for pseudorandom pattern generators were implemented as a watermark generator on the CNN-UM.

With respect to future work and possible new developments the following remarks may serve as a source of inspiration:

- The implementations of PDE-based algorithms on Spatial-temporal Instruction Set Computers allow real-time image processing. From a theoretical point of view the study of the relationship between PDEs and CNN dynamics is very interesting and can be further studied.
- It is important to understand the statistical properties of chaotic signals used in applications, e.g. optimization and random number generation. Therefore the determination of the probabilistic properties of a given type of system and the construction of systems having prescribed properties need to be studied.
- One important question is the bit rate of the proposed RBG which has not been discussed in this thesis. The bit rate of the proposed RBG depends on the normalized quantity τ of the designed chaotic oscillator. This means that the chaotic oscillator must be designed for a high frequency range when a high bit rate is required. Therefore, this work shows again the importance of high frequency designs of chaotic oscillators for applications like this one.

- Statistical tests have shown that the proposed RBG has good statistical properties. However, unpredictability of the system has not been tested yet. More complex attractors such as n-scrolls or scroll grid attractors may further improve unpredictability at this point.

- One possible cryptanalytic opponent technique for the proposed RBG might be synchronization of the chaotic system. The aim is basically then to make a one step ahead prediction of the bits sequence by using an identical chaotic system that is driven by a binary signal coming from the proposed RBG. However, even given a full trajectory of the chaotic signal for reaching synchronization this is a challenge. By using only binary information where a state variable of the trajectory is located as a driver, it is not easy to synchronize two chaotic systems and eventually to predict the next bit.

- 2-D CA for pseudorandom pattern generators show that these are better than 1-D CA and LFSR [36]. However, 2-D CA for PPGs has not been sufficiently analyzed because of lack of applications. Therefore, we believe that their use in watermarking generators will motivate future work in this field.

Bibliography

[1] A. Abel and W. Schwarz. Chaos communications - principles, schemes and system analysis. *Proceeding of the IEEE*, 90(5):691–710, 2002.

[2] K. Aihara, T. Takabe, and M Toyoda. Chaotic neural networks. *Physics Letters A*, 144(6,7):333–340, 1990.

[3] L. Alvarez, F. Guichard, P. L. Lions, and J. M. Morel. Axioms and fundamental equations of image processing. *Arch. Rational Mech. Anal*, 123:199–257, 1993.

[4] D. Angeli. A Lyapunov approach to incremental stability properties. *IEEE Trans. Automatic Control*, 47(3):410–421, 2002.

[5] P. Arena, S. Baglio, L. Fortuna, and G. Manganaro. Chua's circuit can be generated by CNN cells. *IEEE Trans. Circuits and Systems-I*, 42:123–125, 1995.

[6] P. Arena, S. Baglio, L. Fortuna, and G. Manganaro. Generation of n-double scrolls via cellular neural networks. *Int. J. Circuit Theory and Applications*, 24:241–252, 1996.

[7] P. Arena, S. Baglio, L. Fortuna, and G. Manganaro. State controlled CNN: A new strategy for generating high complex dynamics. *IEICE Trans. Fundamentals*, E79-A:1647–1657, 1996.

[8] P. Arena, A. Basile, M. Bucolo, and L. Fortuna. An object oriented segmentation on analog CNN chip. *IEEE Trans. Circuits and Systems-I*, 50(7):837– 846, 2003.

[9] P. Arena, A. Basile, L. Fortuna, M. E. Yalçın, and J. Vandewalle. Watermarking for authentication of video on CNN-UM. In *Proceedings of IEEE Int. Workshop on Cellular Neural Networks and Their Applications*, pages 78–83, Frankfurt, Germany, 2002.

[10] S. Arık and V. Tavşanoğlu. A sufficient condition for absolute stability of a larger class of dynamical neural networks. *IEEE Trans. Circuits and Systems-I*, 47(5):758–760, 2000.

[11] A. Arneodo, P. Coullet, and E. A. Spiegel. Chaos in a finite macroscopic system. *Physical Review Letters*, 92 A(8):369–373, 1982.

[12] A. Arneodo, P. Coullet, and C. Tresser. Possible new strange attractors with spiral structure. *Commun. Math. Phys.*, 79:268–270, 1981.

[13] H. Asai, K. Onodera, T. Kamio, and H. Ninomiya. A study of hopfield neural networks with external noises. In *IEEE International Conference on Neural Networks (ICNN'95)*, volume 4, pages 1584–1589, 1995.

[14] D. P. Atherton. *Nonlinear Control Engineering–Describing Function Analysis and Design.* Van Nostrand Reinhold, New York, 1975.

[15] V. Bagini and M. Bucci. A design of reliable true random number generator for cryptographic applications. In Ç. K. Koç and C. Paar, editors, *Cryptographic Hardware and Embedded Systems, First International Workshop*, volume LCNS 1717 of *Lecture Notes in Computer Science*, pages 204–218. Springer-Verlag, 1999.

[16] D. Balya, B. Roska, E. Nemeth, T. Roska, and F.S. Werblin. A qualitative model framework for spatio-temporal effects in vertebrate retina. In *Proceedings of IEEE Int. Workshop on Cellular Neural Networks and Their Applications*, pages 165–170, Catania, Italy, 2000.

[17] P. Bartissol and L. O. Chua. The double hook. *IEEE Trans. Circuits and Systems*, 35:1512–1522, 1988.

[18] F. Bartolini, A. Tefas, M. Barni, and I. Pitas. Image authentication techniques for surveillance applications. *Proceedings of the IEEE*, 89(9):1403–1418, 2001.

[19] S. Boccaletti, J. Kurths, G. Osipov, D. L. Valladares, and C. S. Zhou. The synchronization of chaotic systems. *Physics Reports*, 366:1–101, 2002.

[20] G. Boffetta, M. Cencini, M. Falcioni, and A. Vulpiani. Predictability : a way to characterize complexity. *Physics Reports*, 356:367–474, 2002.

[21] S. Boyd and C. Barratt. *Linear Controller Design - Limits of Performance.* Prentice-Hall, 1991.

[22] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear matrix inequalities in system and control theory*, volume 15. SIAM (Studies in Applied Mathematics), 1994.

[23] R. Carmona, F. Jimenez-Garrido, R. Dominguez-Castro, S. Espejo, T. Roska, C. Rekecky, I. Petras, and A. Rodriguez-Vazquez. A bio-inspired 2-layer mixed-signal mixed-signal flexible programmable chip for early vision. *IEEE Trans. Neural Networks*, 14(5):1313–1336, 2003.

[24] T. L. Carroll and L. M. Pecora. Synchronizing nonautonomous chaotic circuits. *IEEE Trans. Circuits and Systems-II*, 40(10):646–650, 1993.

[25] M. Casdagli. Nonlinear prediction of chaotic time series. *Physica D*, 35:335–356, 1989.

[26] K. Cattell and S. Zhang. Minimal cost one-dimensional linear hybrid CA of degree through 500. *J. Electronic Testing: Theory and Applications*, 6:255–258, 1995.

[27] B. Chandler, C. Rekeczky, Y. Nishio, and A. Ushida. Using adaptive simulated annealing in CNN template learning—a powerful alternative to genetic algorithms. In *Proceedings of the European Conferance Circuit Theory and Design*, pages 655–660, Budapest, Hungary, September 1997.

[28] G. Chen. *Controlling Chaos and Bifurcations in Engineering Systems.* CRC Press (Boca Raton, USA), 1999.

[29] G. Chen and X. Dong. *From Chaos to Order - Perspectives, Methodologies, and Applications.* World Scientific Pub. Co. Singapore, 1998.

[30] G. Chen and T. Ueta. *Chaos in Circuits and Systems.* World Scientific Pub. Co. Singapore, 2002.

[31] G. Chen and X. Yu. *Chaos control : Theory and Applications*, volume 292 of *Lecture Notes in Control and Information Sciences.* Springer-Verlag, 2003.

[32] H. F. Chen and J. M. Liu. Open-loop chaotic synchronization of injection-locked semiconductor lasers with gigahertz range modulation. *IEEE Journal Quantum Electronics*, 36(1):27–34, 2000.

[33] J. Chen. On computing the maximal delay intervals for stability of linear delay systems. *IEEE Trans. Automatic Control*, 40:1087–1093, 1995.

[34] J. Chen and H. A. Latchman. Frequency sweeping tests for stability independent of delay. *IEEE Trans. Automatic Control*, 40:1640–1645, 1995.

[35] L. Chen and K. Aihara. Chaotic simulated annealing by a neural network model with transient chaos. *Neural Networks*, 8(6):915–930, 1995.

[36] D. R. Chowdhury, I. S. Gupta, and P. P. Chaudhuri. A class of two-dimensional cellular automata and applications in random pattern testing. *J. Electronic Testing: Theory and Applications*, 5:65–80, 1994.

[37] L. O. Chua. The genesis of Chua's circuit. *Archiv für Elektronik und Übertragungstechnik*, 46(4):250–257, 1992.

[38] L. O. Chua. Chua's circuit 10 years later. *Int. J. Circuit Theory and Applications*, 22:279–305, 1994.

[39] L. O. Chua. Special issue on nonlinear waves, patterns and spatio-temporal chaos. *IEEE Trans. Circuits and Systems-I*, 42:557–558, 1995.

[40] L. O. Chua. *CNN: a Paradigm for Complexity.* World Scientific Pub. Co Singapore, 1998.

[41] L. O. Chua, M. Hasler, S. Moschytz, and J. Neirynck. Autonomous cellular neural networks : A unified paradigm for pattern formation and active wave propagation. *IEEE Trans. Circuits and Systems-I*, 42(10):559–577, 1995.

[42] L. O. Chua, L. Kocarev, K. Eckert, and M. Itoh. Experimental chaos synchronization in Chua's circuit. *Int. J. Bifurcation and Chaos*, 2:705–708, 1992.

[43] L. O. Chua, M. Komuro, and T. Matsumoto. The double scroll family. *IEEE Trans. Circuits and Systems*, 33:1072–1118, 1986.

[44] L. O. Chua and G. N. Lin. Canonical realization of Chua's circuit family. *IEEE Trans. Circuits and Systems-I*, 37:885 –902, 1990.

[45] L. O. Chua and T. Roska. The CNN paradigm. *IEEE Trans. Circuits and Systems-I*, 40(3):147–156, 1993.

[46] L. O. Chua and T. Roska. *Cellular neural networks and visual computing - Foundations and applications.* Cambridge University, 2001.

[47] L. O. Chua, C. W. Wu, A. Huang, and Z. Q. Zhong. A universal circuit for studying and generating chaos - Part I: Routes to chaos & Part II: Strange attractors. *IEEE Trans. Circuits and Systems-I*, 4(2):745–761, 1993.

[48] L. O. Chua and L. Yang. Cellular neural networks: Theory and applications. *IEEE Trans. Circuits and Systems*, 35(10):1257–1290, 1988.

[49] P. P. Civalleri and M. Gilli. On stability of cellular neural networks. *Journal of VLSI Signal Processing*, 23(2/3):429–435, 1999.

[50] P. A. Cook. *Nonlinear Dynamical Systems.* Prentice-Hall, London, 1986.

[51] F. Corinto and M. Gilli. Comparison between the dynamic behaviour of Chua-Yang and full-range cellular neural networks. *Int. J. Circuit Theory and Applications*, 31(5):423–441, 2003.

[52] Intel Corporation. The Intel random number generator. Technical report, Intel Corporation, 1999.

[53] P. Coullet, C. Tresser, and A. Arneodo. Transition to stochasticity for a class of forced oscillators. *Physical Review Letters*, 72 A(4,5):268–270, 1979.

[54] K. R. Crounse and L. O. Chua. The CNN universal machine is a turing machine. *IEEE Trans. Circuits and Systems-I*, 43(4):353–355, 1996.

[55] K. R. Crounse, E. L. Fung, and L. O. Chua. Efficient implementation of neighborhood logic for cellular automata via the cellular neural network universal machine. *IEEE Trans. Circuits and Systems-I*, 44(4):355–361, 1997.

[56] K. R. Crounse, T. Yang, and L. O. Chua. Pseudo-random sequence generation using the cnn universal machine with application to cryptography. In *Proceedings of IEEE Int. Workshop on Cellular Neural Networks and Their Applications*, pages 133–138, Sevilla, Spain, 1996.

[57] J. M. Cruz and L. O. Chua. A CMOS IC nonlinear resistor for Chua's circuit. *IEEE Trans. Circuits and Systems-I*, 39:985–995, 1992.

[58] J. M. Cruz and L. O. Chua. Design of high-speed, high-density CNNs in CMOS technology. *Int. J. Circuit Theory and Applications*, 20:555–572, 1992.

[59] M. Csapodi, J. Vandewalle, and B. Preneel. CNN algorithms for video authentication and copyright protection. *The Journal of VLSI Signal Processing-Systems for Signal, Image, and Video Technology*, 23(2-3):449–463, 1999.

[60] M. Csapodi, J. Vandewalle, and T. Roska. High speed calculation of cryptographic hash functions by CNN chips. In *Proceedings of IEEE Int. Workshop on Cellular Neural Networks and Their Applications*, pages 186–191, London, England, 1998.

[61] G. Cserey, C. Rekeczky, and P. Foldesy. PDE-based histogram modification with embedded morphological processing of the level-sets. *Journal of Circuits, Systems, and Computers*, 12(4):519–538, 2003.

[62] K. M. Cuomo and A. V. Oppenheim. Circuit implementation of synchronized chaos with application to communications. *Physical Review Letters*, 71(1):65–68, 1993.

[63] P. F. Curran and L. O. Chua. Absolute stability theory and synchronization problem. *Int. J. Bifurcation and Chaos*, 7(6):1375–1382, 1997.

[64] P. F. Curran, J. A. K. Suykens, and L. O. Chua. Absolute stability theory and master-slave synchronization. *Int. J. Bifurcation and Chaos*, 7(12):2891–2896, 1997.

[65] F. Dachselt and W. Schwarz. Chaos and cryptography. *IEEE Trans. Circuits and Systems-I*, 48(12):1498–1509, 2001.

[66] D. Davis, R. Ihaka, and P. Fenstermacher. Cryptographic randomness from air turbulence in disk drives. In Y. G. Desmedt, editor, *Advances in Cryptology, 14th Annual International Cryptology Conference- CRYPTO '94*, volume LNCS 0839 of *Lecture Notes in Computer Science*, pages 114–120. Springer-Verlag, 1994.

[67] H. Dedieu, M. P. Kennedy, and M. Hasler. Chaos shift keying: modulation and demodulation of a chaotic carrier using self-synchronizing Chua's circuits. *IEEE Trans. Circuits and Systems-II*, 40(10):634–642, 1993.

[68] M. Delgado-Restituto and A. Rodriguez-Vasquez. A CMOS monolithic Chua's circuit. *J. Circuit Syst. Comput.*, 3(2), 1993.

[69] P. Deregel. Chua's oscillator : A zoo of attractor. In R. N. Madan, editor, *Chua's circuit: A Paradigm for Chaos*, volume 1 of *World Scientific Series on Nonlinear Science, Series B*, pages 179–229. World Scientific, 1993.

[70] R. Dominguez-Castro, R. Carmona, S. Espejo, and A. Rodriguez-Vazquez. A 20 × 22 CNN-UM chip with on-chip optical sensors. In T. Roska and A. Rodriguez-Vazquez, editors, *Towards the Visual MicroProcessor - VLSI Design and the use of Cellular Network Universal Machines*, chapter 6.1. John Wiley & Sons, 2001.

[71] R. Dominguez-Castro, S. Espejo, A. Rodriguez-Vazquez, and R. A. Carmona. A CNN universal chip in CMOS technology. In *Proceedings of IEEE Int. Workshop on Cellular Neural Networks and Their Applications*, pages 91–96, Rome, Italy, 1994.

[72] J. Doyne and J. J. Sidorowich. Predicting chaotic time series. *Physical Review Letters*, 59(8):845–848, 1987.

[73] D. Eastlake, S. Crocker, and J. Schiller. RFC 1750: randomness recommendations for security. Technical report, Network Working Group, December 1994.

[74] R. Elson, A. I. Selverston, R. Huerta, M. I. Rabinovich, and H. D. I. Abarbanel. Synchronous behavior of two coupled biological neurons. *Physical Review Letters*, 81:5692–5695, 1998.

[75] A. S. Elwakil and M. P. Kennedy. High frequency Wien-type chaotic oscillator. *Electronics Letters*, 34(12):1161–1162, 1998.

[76] A. S. Elwakil and M. P. Kennedy. Construction of classes of circuit-independent chaotic oscillators using passive-only nonlinear devices. *IEEE Trans. Circuits and Systems-I*, 48(3):289–307, 2001.

[77] A. S. Elwakil, K. N. Salama, and M. P. Kennedy. A system for chaos generation and its implementation in monolithic form. In *Proceedings of the IEEE Int. Symp. Circuits and Systems (ISCAS'2000)*, volume V, pages 217–220, 2000.

[78] A.S. Elwakil and M. P. Kennedy. Improved implementation of Chua's chaotic oscillator using current feedback opamp. *IEEE Trans. Circuits and Systems-I*, 47(1):76–79, 2000.

[79] S. Espejo, R. Carmona, R. Dominguez-Castro, and A. Rodriguez-Vazquez. A VLSI oriented continuous time CNN model. *Int. J. Circuit Theory and Applications*, 24:341–356, 1996.

[80] A. Fabre. Insensitive voltage-mode and current-mode filters from commercially available transimpedance op-amps. *IEE Proceedings-G*, 140(319-321):319–321, 1993.

[81] R. C. Fairfield, R. L. Mortenson, and K. B. Coulthart. An LSI random number generator (RNG). In G.R. Blakley and David Chaum, editors, *Advances in Cryptology,Proceedings of CRYPTO 84*, volume LNCS 0196 of *Lecture Notes in Computer Science*, pages 203–230. Springer-Verlag, 1987.

[82] R. Fletcher. *Practical Methods of Optimization.* Wiley Chichester, 1987.

[83] P. Foldesy, L. Kek, A. Zarandy, and G. Bartfai T. Roska. Fault-tolerant design of analogic CNN templates and algorithms-Part I: The binary output case. *IEEE Trans. Circuits and Systems-I*, 46(2):312–322, 1999.

[84] A. M. Fraser. Information and entropy in strange attractors. *IEEE Transactions on Information Theory*, 35(2):245–262, 1989.

[85] G. L. Friedman. The trustworthy digital camera: restoring credibility to the photographic image. *IEEE Trans. on Consumer Electronics*, 39:905–910, 1993.

[86] T. Fujiwara, Y. Horio, and K. Aihara. An integrated multi-scroll circuit with floating-gate MOSFETs. In *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS'03)*, volume III, pages 180–183, Bangkok, Thailand, 2003.

[87] V. Gal, S. Grun, and R. Tetzlaff. Analyzing multidimensional neural activity via CNN-UM. In *Proceedings of IEEE Int. Workshop on Cellular Neural Networks and Their Applications*, pages 243–250, Frankfurt, Germany, 2002.

[88] S.B. Gelfand and S.K. Mitter. Metropolis-type annealing algorithms for global optimization in $\mathbb{R}^d$. *SIAM Journal on Control and Optimization*, 31(1):111–131, 1993.

[89] A. Genesio and A. Tesi. Chaos prediction in nonlinear feedback systems. *IEE Proceedings-D*, 134(4):313–320, 1991.

[90] R. Genesio and A. Tesi. A harmonic balance approach for chaos prediction: Chua's circuit. *Int. J. Bifurcation and Chaos*, 1(2):61–79, 1992.

[91] R. Genesio and A. Tesi. Harmonic balance methods for the analysis of chaotic dynamics in nonlinear systems. *Automatica*, 28(3):531– 548, 1992.

[92] R. Genesio and A. Tessi. Distortion control of chaotic system: the Chua's circuit. In R. N. Madan, editor, *Chua's circuit: A Paradigm for Chaos*, volume 1 of *World Scientific Series on Nonlinear Science, Series B*, pages 514–534. World Scientific, 1993.

[93] M. Gilli, T. Roska, L. O. Chua, and P. P. Civalleri. On the relationship between CNNs and PDEs. In *Proceedings of IEEE Int. Workshop on Cellular Neural Networks and Their Applications*, pages 16–24, Frankfurt, Germany, 2002.

[94] J. A. González, L. I. Reyes, J. J. Suarez, L. E. Guerrero, and G. Gutiérrez. A mechanism for randomness. *Physics Letters A*, 295(1):25–34, 2002.

[95] M. Götz, U. Feldmann, and W. Schwarz. Synthesis of higher-dimensional Chua circuits. *IEEE Trans. Circuits and Systems-I*, 40(11):854–860, 1993.

[96] M. Gotz, K. Kelber, and W. Schwarz. Discrete-time chaotic encryption systems. Part I. statistical design approach. *IEEE Trans. Circuits and Systems-I*, 44(10):963–970, 1997.

[97] J. Guckenheimer. Toolkit for nonlinear dynamics. *IEEE Trans. Circuits and Systems*, 30(8):586–590, 1983.

[98] M. Guide. Concept for a high-performance random number generator based on physical random phenomena. *Frequenz*, 39:187–190, 1985.

[99] C. Güzeliş. Chaotic cellular neural networks made of Chua's circuits. In R. N. Madan, editor, *Chua's circuit: A Paradigm for Chaos*, volume 1 of *World Scientific Series on Nonlinear Science, Series B*, pages 952–961. World Scientific, 1993.

[100] C. Güzeliş and L. O. Chua. Stability analysis of generalized cellular neural networks. *Int. J. Circuit Theory and Applications*, 21:1–33, 1993.

[101] C. Güzeliş, S. Karamahmut, and I. Genç. A recurrent perceptron learning algorithm for cellular neural networks. *ARI - Interdiscip. Journal of Phys. and Eng. Sciences*, 51(4):296–309, 1999.

[102] J. Hale. *Functional differential equations*. New York: Academic, 1977.

[103] K. S. Halle, C. W. Wu, M. Itoh, and L. O. Chua. Spread spectrum communication through modulation of chaos in Chua's circuit. *Int. J. Bifurcation and Chaos*, 3(2):469–477, 1993.

[104] F. Han, X. Yu, G. Chen, W. Liu, and Y. Feng. Generation of multi-scroll chaos using second-order linear systems with hysterresis. In *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS'03)*, volume III, pages 84–87, Bangkok, Thailand, 2003.

[105] T. S. Han and M. Hoshi. Interval algorithm for random number generation. *IEEE Transactions on Information Theory*, 43(2):599–611, 1997.

[106] M. Hanggi and G. S. Moschytz. An exact and direct analytical method for the design of optimally robust CNN templates. *IEEE Trans. Circuits and Systems-I*, 46(2):304–311, 1999.

[107] H. Harrer and J. Nossek. Discrete-time cellular neural networks. *Int. J. Circuit Theory and Applications*, 20(5):453–467, 1992.

[108] M. Hasegawa, T. Ikeguchi, T. Matozaki, and K. Aihara. An analysis on additive effects of nonlinear dynamics for combinatorial optimization. *IEICE Trans. Fundementals*, E80-A(1):206–213, 1997.

[109] M. Hasler, Y. Maistrenko, and O. Popovich. Simple example of partial synchronization of chaotic systems. *Physical Review E*, 58:6843–6846, 1998.

[110] M. Hasler, G. Mazzini, M. Ogorzalek, R. Rovatti, and G. Setti. Special issue on applications of nonlinear dynamics to electronic and information engeneering. *Proceedings of IEEE*, 90(5):631–638, 2002.

[111] Y. Hayakawa, A. Marumoto, and Y. Sawada. Effects of the chaotic noise on the performance of a neural network model for optimization problems. *Physical Review E*, 51(4):R2693–R2696, 1995.

[112] Y. He. Chaotic simulated annealing with decaying chaotic noise. *IEEE Trans. Neural Networks*, 13(6):1526–1531, 2002.

[113] D. J. Hill and P. J. Moylan. The stability of nonlinear dissipative systems. *IEEE Trans. Automatic Control*, 21:709–711, 1976.

[114] J.J. Hopfield and D.W. Tank. Neural computation of decisions in optimization problems. *Biological Cybernetics*, 52:1–25, 1985.

[115] P. D. Hortensius, R. D. McLeod, W. Pries, D. M. Miller, and H. C. Card. Cellular automata-based pseudorandom number generators for built-in self-test. *IEEE Trans. Computer-Aided Design*, 8(8):842–859, 1989.

[116] M. Jakobsson, E. Shriver, B. K. Hillyer, and A. Juels. A practical secure physical random bit generator. In *Proceedings of The Fifth ACM Conference on Computer and Communications Security*, pages 103 – 111, San Francisco, California, United States, November 1998.

[117] A. Juels, M. Jakobsson, E. Shriver, and B. K. Hillyer. How to turn loaded dice into fair coins. *IEEE Transactions on Information Theory*, 46(3):911–921, 2000.

[118] B. Jun and P. Kocher. The Intel random number generator. Technical report, White Paper Prepared for Intel Corporation, Cryptography Research Inc., 1999. Available for free download : `http://www.intel.com/design/security/rng/CRIwp.htm`.

[119] E. W. Kamen. Linear systems with commensurate time delays: stability and stabilization independent of delay. *IEEE Trans. Automatic Control*, 27:367–375, 1982.

[120] E. W. Kamen. Correction to linear systems with commensurate time delays: stability and stabilization independent of delay. *IEEE Trans. Automatic Control*, 28:248–249, 1983.

[121] H. Kantz. A robust method to estimate the maximal lyapunov exponent of a time series. *Physics Letters A*, 185:77, 1994.

[122] T. Kapitaniak. Synchronization of chaos using continuous control. *Physical Review E*, 50(2):1642–1644, 1994.

[123] T. Kapitaniak and L. O. Chua. Hyperchaotic attractors of unidirectionally-coupled Chua's circuits. *Int. J. Bifurcation and Chaos*, 4(1):477–482, 1994.

[124] T. Kapitaniak, L. O. Chua, and G. Q. Zhong. Experimental hyperchaos in coupled Chua's circuits. *IEEE Trans. Circuits and Systems-I*, 41(7):499–403, 1994.

[125] M. Kataoka and T. Saito. A $4-d$ chaotic oscillator with a hysteresis 2–port VCCS. In *Proceedings of IEEE Int. Symp. on Circuits and Systems (ISCAS 2000)*, volume V, pages 418–421, 2000.

[126] F. Kavaslar and C. Güzeliş. A computer-assisted investigation of a $2D$ array of Chua's circuit. *IEEE Trans. Circuits and Systems-I*, 42:721–735, 1995.

[127] J. Kawate, Y. Nishio, and A. Ushida. On synchronization phenomena in chaotic systems coupled by transmission line. In *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS'2000)*, volume III, pages 479–482, Geneva, Switzerland, 2000.

[128] L. Kek and A. Zarandy. Implementation of large-neighbourhood nonlinear templates on the CNN universal machine. *Int. J. Circuit Theory and Applications*, 26(6):551–566, 1998.

[129] M. P. Kennedy. Robust op amp implementation of Chua's circuit. *Frequenz*, 46:66–80, 1992.

[130] M. P. Kennedy. Synthesis of continuous piecewise-linear resistors for Chua's circuit family using operational amplifiers, diodes and linear resistors. *Int. J. Circuit Theory and Applications*, 21(5), 1993.

[131] M. P. Kennedy. Three steps to chaos. II. a Chua's circuit primer. *IEEE Trans. Circuits and Systems-I*, 40(10):657–674, 1993.

[132] M. P. Kennedy. Communicating with chaos state of the art and engineering challenges. In *Proceedings of the* 4^{th} *Int. Workshop in Nonlinear Dynam. Electron. Syst.(NDES'96)*, pages 1–8, Sevilla, Spain, 1996.

[133] M. P. Kennedy and M. J. Ogorzalek. Special issue on chaos synchronization and control: theory and applications. *IEEE Trans. Circuits and Systems-I*, 44(10):853–855, 1997.

[134] H. K. Khalil. *Nonlinear Systems.* Macmillan Publishing Company, New York, 1992.

[135] A. I. Khibnik, D. Roose, and L. O. Chua. On periodic orbits and homoclinic bifurcations in Chua's circuit with a smooth nonlinearity. In R. N. Madan, editor, *Chua's circuit: A Paradigm for Chaos*, volume 1 of *World Scientific Series on Nonlinear Science, Series B*, pages 145–178. World Scientific, 1993.

[136] P. Kinget and M. Steyaert. Evaluation of CNN template robustness toward VLSI implementation. *Int. J. Circuit Theory and Applications*, 24(1):93–110, 1996.

[137] D. E. Knuth. *The Art of Computer Programming.* Addison- Wesley, 1969.

[138] L. Kocarev. Chaos-based cryptography: a brief overview. *IEEE Circuits and Systems Magazine*, 1(3):6–21, 2001.

[139] L. Kocarev, K. S. Halle, K. Eckert, and L. O. Chua. Experimental demonstration of secure communications via chaotic synchronization. *Int. J. Bifurcation and Chaos*, 2:709–713, 1992.

[140] L. Kocarev, L. Karadzinov, and L. O. Chua. n-dimensional canonical Chua's circuit. *J. Circuit Syst. Comput.*, 3(1):239–258, 1993.

[141] L. Kocarev and U. Parlitz. Generalized synchronization, predictability, and equivalence of unidirectionally coupled dynamical systems. *Physical Review Letters*, 76(11):1816–1819, 1996.

[142] L. Kocarev, U. Parlitz, and R. Brown. Robust synchronization of chaotic systems. *Physical Review E*, 61(4):3716–3720, 2000.

[143] L. Kocarev and T. Roska. Dynamics of the Lorenz equation and Chua's equation : A tutorial. In R. N. Madan, editor, *Chua's circuit: A Paradigm for Chaos*, volume 1 of *World Scientific Series on Nonlinear Science, Series B*, pages 25–55. World Scientific, 1993.

[144] T. Kohda and A. Tsuneda. Statistics of chaotic binary sequences. *IEEE Transactions on Information Theory*, 43:104–112, 1997.

[145] R. Koike, H. Sekiya, N. Miyabayashi, S. Moro, and S. Mori. Synchronization of two chaotic circuits coupled by delay line. In *Proceedings of European Conference on Circuits Theory and Design (ECCTD'97)*, volume 3, pages 1280–1285, Budapest, Hungary, 1997.

[146] G. Kolumban and M. P. Kennedy. The role of synchronization in digital communications using chaos-Part III: Performance bounds for correlation receivers. *IEEE Trans. Circuits and Systems-I*, 47(12):1673–1683, 2000.

[147] G. Kolumban, M. P. Kennedy, and L. O. Chua. The role of synchronization in digital communications using chaos-Part I: Fundamentals of digital chaotic communications. *IEEE Trans. Circuits and Systems-I*, 44(10):927–936, 1997.

[148] G. Kolumban, M. P. Kennedy, and L. O. Chua. The role of synchronization in digital communications using chaos-Part II: Chaotic modulation of digital communications. *IEEE Trans. Circuits and Systems-I*, 45(11):1129–1140, 1998.

[149] G. Kolumban, M. P. Kennedy, and G. Kis. Performance improvement of chaotic communications systems. In *Proceedings of* 13^{th} *European Conference on Circuit Theory and Design (ECCDT'97)*, pages 284–289, Budapest, Hungary, 1997.

[150] G. Kolumban, G. Kis, Z. Jako, and M. P. Kennedy. FM-DCSK: a robust modulation scheme for chaotic communications. *IEICE Transactions*, E81-A(9):1798–1802, 1998.

[151] G. Kolumban, B. Vizvari, W. Schwarz, and A. Abel. Differential chaos shift keying: A robust coding for chaotic communication. In *Proceedings of the* 4^{th} *Int. Workshop in Nonlinear Dynam. Electron. Syst.(NDES'96)*, pages 87–92, Sevilla, Spain, 1996.

[152] M. Komuro, R. Tokunaga, T. Matsumoto, L. O. Chua, and A. Hotta. Global bifurcation analysis of the double scroll circuit. *Int. J. Bifurcation and Chaos*, 1(1):139–182, 1991.

[153] T. Kozek, T. Roska, and L. O. Chua. Genetic algorithm for cnn template learning. *IEEE Trans. Circuits and Systems-I*, 40(3):392–402, 1993.

[154] N. N. Krasovskii and J. L. Brenner. *Stability of motion: applications of Lyapunov's second method to differential Systems and equations with delay.* Stanford university press Stanford (Calif.), 1963.

[155] T. Kwok and K. A. Smith. A unified framework for chaotic neural-network approaches to combinatorial optimization. *IEEE Trans. Neural Networks*, 10(4):978–981, 1999.

[156] V. Lakshmikantham, D. D. Bainov, and P. S. Simeonov. *Theory of Impulsive Differential Equations.* World Scientific, Singapore, 1998.

[157] G. C. Langelaar and I. Setyawan. Watermarking digital image and video data: A state-of-art overview. *IEEE Signal Processing Magazine*, 9:21–46, 2000.

[158] A. Lasota and M. C. Mackey. *Chaos, Fractal and Noise*, volume 97 of *Applied Mathematical Sciences.* Springer, 1994.

[159] X. Liao and G. Chen. Chaos synchronization of general Lur'e systems via time-delay feedback control. *Int. J. Bifurcation and Chaos*, 13(1):207–213, 2003.

[160] E. T. Lin and E. J. Delp. A review of fragile image watermarks. In *Proceedings of the Multimedia and Security Workshop Multimedia contents*, pages 25–29, Orlando, USA, 1999.

[161] G. Linan, S. Espejo, R. Dominguez-Castro, and A. Rodriguez-Vazquez. ACE4k: an analog I/O 64x64 visual microprocessor chip with 7-bit analog accuracy. *Int. J. Circuit Theory and Applications*, 30:89–116, 2002.

[162] G. Linan, S. Espejo, R. Dominguez-Castro, and A. Rodriguez-Vazquez. Architectural and basic circuit considerations for a flexible 128 × 128 mixed-signal SIMD vision chip. *Analog Integrated Circuits and Signal Processing*, 33:179–190, Nov. 2002.

[163] G. Linan, A. Rodriguez-Vazquez, S. Espejo, and R. Dominguez-Castro. ACE16k: A 128x128 focal plane analog processor with digital I/O. In *Proceedings of IEEE Int. Workshop on Cellular Neural Networks and Their Applications*, pages 132–139, Frankfurt, Germany, 2002.

[164] D. Lind and B. Marcus. *An Introduction to Symbolic Dynamics and Coding.* Cambridge, 1995.

[165] J. M. Maciejowski. *Multivariable feedback design.* Addison-Wesley, 1989.

[166] R. N. Madan. *Chua's Circuit: A paradigm for chaos.* World Scientific, Singapore, 1993.

[167] R. N. Madan. (Guest Editor). Special issue on Chua's circuit: A paradigm for chaos, Part I. *J. Circuit Syst. Comput.*, 3(1), 1993.

[168] M. Maes, T. Kalker, J-P. M. G. Linnartz, J. Talstra, G. F. G. Depovere, and J. Haitsma. Digital watermarking for dvd video copy protection: What issues play a role in designing an effective system? *IEEE Signal Processing Magazine*, 9:47–57, 2000.

[169] D. P. Maher and R. J. Rance. Random number generators founded on signal and information theory. In Ç. K. Koç and C. Paar, editors, *Cryptographic Hardware and Embedded Systems, First International Workshop*, volume 1717 of *Lecture Notes in Computer Science*, pages 219–230. Springer, 1999.

[170] M. S. Mahmoud. *Robust control and filtering for time-delay Systems.* Marcel Dekker, 2000.

[171] G. Manganaro, P. Arena, and L. Fortuna. *Cellular Neural Networks Chaos, Complexity and VLSI Processing.* Springer-Verlag, Berlin, Heidelberg, 1999.

[172] G. Marsalgia. *Diehard.* `http://stat.fsu.edu/~geo/diehard.html`.

[173] T. Matsumoto. A chaotic attractor from Chua's circuit. *IEEE Trans. Circuits and Systems*, 31:1055–1058, 1984.

[174] T. Matsumoto, L. Chua, and M. Komuro. The double scroll. *IEEE Trans. Circuits and Systems*, 32:798–817, 1985.

[175] T. Matsumoto, L. O. Chua, and K. Kobayashi. Hyperchaos: Laboratory experiment and numerical confirmation. *IEEE Trans. Circuits and Systems*, 33(11):1143–1147, 1986.

[176] T. Matsumoto, L. O. Chua, and K. Tokumasu. Double scroll via a two-transistor circuit. *IEEE Trans. Circuits and Systems*, 33:828–835, 1986.

[177] A. Mees and P. Chapman. Homoclinic and heteroclinic orbits in the double scroll attractor. *IEEE Trans. Circuits and Systems*, 34(9):1115–1120, 1987.

[178] N. Memon and W. Wong. Protecting digital media content. *Communications of ACM*, 41(7):35–43, 1988.

[179] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone. *Handbook of Applied Cryptography.* CRC Press, 1997. Available for free download : `http://www.cacr.math.uwaterloo.ca/hac/`.

[180] B. Mirzai, D. Lim, and G. S. Moschytz. Robust CNN templates: Theory and simulations. In *Proceedings of IEEE Int. Workshop on Cellular Neural Networks and Their Applications*, pages 393–398, Sevilla, Spain, 1996.

[181] Ö. Morgül and E. Solak. On the observer based synchronization of chaotic systems. *Physical Review E*, 54(5), 1996.

[182] T. Mori. Criteria for asymptotic stability of linear time-delay systems. *IEEE Trans. Automatic Control*, 30(2):158–161, 1985.

[183] T. Mori and T. Kokame. Stability of $\dot{x}(t) = Ax(t) + Bx(t-\tau)$. *IEEE Trans. Automatic Control*, 34(4):460–462, 1989.

[184] T. Mori, E. Noldus, and M. Kuwahara. A way to stabilize linear systems with delayed state. *Automatica*, 19(5):571–573, 1983.

[185] A. P. Munuzuri, V. P. Munuzuri, M. G. Gesteria, L. O. Chua, and V. P. Villar. Spatiotemporal structures in discretely-coupled arrays of nonlinear circuits: A review. *Int. J. Bifurcation and Chaos*, 5(1):17–50, 1995.

[186] H. Nakano and T. Saito. Basic dynamics from a pulse-coupled network of autonomous integrate-and-fire chaotic circuits. *IEEE Trans. Neural Networks*, 13(1):92–100, 2002.

[187] H. Nijmeijer. Control of chaos and synchronization. *Sys. Control Lett.*, 31(5):259–262, 1997.

[188] H. Nijmeijer. A dynamical control view on synchronization. *Physica D*, 154:219–228, 2001.

[189] H. Nijmeijer and M. Y. Mareels. An observer looks at synchronization. *IEEE Trans. Circuits and Systems-I*, 44(10), 1997.

[190] J. Nocedal and S. J. Wright. *Numerical Optimization.* Springer, 1999.

[191] J. A. Nossek. Design and learning with cellular neural networks. *Int. J. Circuit Theory and Applications*, 24:15–24, 1996.

[192] S. Ozoğuz, A. S. Elwakil, and K. N. Salama. n-scroll chaos generator using nonlinear transconductor. *Elecronics Letters*, 38(14):685–686, 2002.

[193] S. Ozoğuz and N. S. Sengör. On the realization of NPN-only log-domain chaotic oscillators. *IEEE Trans. Circuits and Systems-I*, 50(2):291–294, 2003.

[194] A. Paasio, A. Dawidzuik, K. Halonen, and V. Porra. Minimum size $0.5\mu m$ CMOS programmable 48×48 CNN test chip. In *Proceedings of the European Conferance on Circuit Theory and Design*, pages 154–156, Budapest, Hungary, September 1997.

[195] T. S. Parker and L. O. Chua. *Practical Numerical Algorithms for Chaotic Systems.* Springer-Verlag, 1989.

[196] U. Parlitz, L. Kocarev, T. Stojanovski, and H. Preckel. Encoding messages using chaotic synchronization. *Physical Review E*, 53(4351-4361), 1996.

[197] L. M. Pecora and T. L. Carroll. Synchronization in chaotic systems. *Physical Review Letters*, 64(1):821–823, 1990.

[198] I. Petras, T. Roska, and L. O. Chua. New spatial-temporal patterns and the first programmable on-chip bifurcation test bed. *IEEE Trans. Circuits and Systems-I*, 50(5):619–633, 2003.

[199] A. S. Pikovsky, M. G. Rosenblum, and J. Kurths. *Synchronization - A universal concept in nonlinear sciences.* Cambridge University Press, Cambridge, 2001.

[200] E. Polak and Y. Wardi. Nondifferentiable optimization algorithm for designing control systems having singular value inequalities. *Automatica*, 18(3):267–283, 1982.

[201] B. Preneel. *Analysis and Design of Cryptographic Hash Functions.* PhD thesis, K. U. Leuven, 1993.

[202] A. G. Radwan, A. M. Soliman, and A. El-Sedeek. MOS realization of the double-scroll-like chaotic equation. *IEEE Trans. Circuits and Systems-I*, 50(2):285–288, 2003.

[203] C. Rekeczky. *Dynamic Spatio-temporal Nonlinear Filtering and Detection on CNN Architecture: Theory, Modeling and Applications.* PhD thesis, Analogical and Neural Computing Laboratory, Computer and Automation Institute of the Hungarian Academy of Sciences, Budapest, Hungary, 1999.

[204] A. Rodriguez-Vazquez, S. Espejo, R. Dominguez-Castron, J. L. Huertas, and E. Sanchez-Sinencio. Current-mode techniques for the implementation of continuous- and discrete-time cellular neural networks. *IEEE Trans. Circuits and Systems-II*, 40(3):132–146, 1993.

[205] B. Roska and F.S. Werblin. Vertical interactions across ten parallel, stacked representations in the mammalian retina. *Nature*, 410:583–587, 2001.

[206] T. Roska. Computer-sensor: Spatial-temporal computers for analog array signals, dynamically integrated with sensor. *Journal of VLSI Signal Processing*, 23(2/3):221–237, 1999.

[207] T. Roska. Computational and computer complexity of analogic cellular wave computer. In *Proceedings of IEEE Int. Workshop on Cellular Neural Networks and Their Applications*, pages 323–338, Frankfurt, Germany, 2002.

[208] T. Roska. Computational and computer complexity of analogic cellular wave computers. *Journal of Circuits, Systems, and Computers*, 12(4):539–562, 2003.

[209] T. Roska. SENSIT project proposal. Technical report, Computer and Automation Research Institute of the Hungarian Academy of Sciences, Budapest, Hungary, 2003.

[210] T. Roska, T. Boros, A. Radvanyi, P. Thiran, and L. O. Chua. Detecting moving and standing objects using cellular neural network. *Int. J. Circuit Theory and Applications*, 20(5):613–628, 1992.

[211] T. Roska and L. O. Chua. Cellular neural networks with nonlinear and delay-type template elements and non-uniform grids. *Int. J. Circuit Theory and Applications*, 20(5):469–482, 1992.

[212] T. Roska and L. O. Chua. The CNN universal machine: an analogic array computer. *IEEE Trans. Circuits and Systems-I*, 40(3):163–173, 1993.

[213] T. Roska, L. O. Chua, D. Wolf, T. Kozek, R. Tetzlaff, and F. Puffer. Simulating nonlinear waves and partial differential equations via CNN. I. basic techniques. *IEEE Trans. Circuits and Systems-I*, 42(10):807–815, 1995.

[214] T. Roska and A. Rodriguez-Vazquez. Towards visual microprocessors. *Proceedings of the IEEE*, 90(7):1244–1257, 2002.

[215] T. Roska, A. Rodriguez-Vazquez, and J. Vandewalle. Theory, design and applications of cellular neural networks: Part II: Design and applications. *Int. J. Circuit Theory and Applications*, 27(1):1 – 207, 1999.

[216] T. Roska and J. Vandewalle, editors. *Cellular Neural Networks.* John Wiley & Sons, 1994.

[217] T. Roska, A. Zarandy, S. Zold, P. Foldesy, and P. Szolgay. The computational infrastructure of analogic CNN computing. I. the CNN-UM chip prototyping system. *IEEE Trans. Circuits and Systems-I*, 46(2):261–268, 1999.

[218] O. E. Rössler. Chaotic oscillations-an example of hyperchaos. *Nonlinear Oscillations in Biology, Lectures in Applied Mathematics*, 17:141–156, 1979.

[219] R. Rovatti and G. Mazzini. Tensor function analysis of quantized chaotic piecewise-affine pseudo-markov systems. I. second-order correlations and self similarity. *IEEE Trans. Circuits and Systems-I*, 49(2):137–149, 2002.

[220] R. Rovatti, G. Setti, and G. Mazzini. Chaotic complex spreading sequences for asynchronous DS-CDMA. *IEEE Trans. Circuits and Systems-I*, 45(4):496–506, 1998.

[221] T. Saito. An approach toward higher dimensional hysteresis chaos generators. *IEEE Trans. Circuits and Systems-I*, 37(3):399–409, 1990.

[222] K. N. Salama, S. Ozoğuz, and A. S. Elwakil. Generation of n-scroll chaos using nonlinear transconductors. In *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS'03)*, volume III, pages 176–179, Bangkok, Thailand, 2003.

[223] S. Sastry. *Nonlinear Systems. Analysis, Stability, and control.* Interdisciplinary Applied Mathematics. Systems and Control. Springer, 1999.

[224] F. A. Savacı, S. Gunel, N. Ozkurt, and I. I. Yapali. Harmonic balance analysis of the generalized chua's circuit. In *Progress In Nonlinear Science, Int. Conference dedicated to the 100th Anniversary of A. A. Andronov*, pages 306–307, N. Novgorod, Russia, July 2-6 2001.

[225] F. A. Savacı and J. Vandewalle. On the stability analysis of cellular neural networks. *IEEE Trans. Circuits and Systems-I*, 40(3):213 –215, 1993.

[226] F. A. Savacı and M. E. Yalçın. Analysis of chaotic state modulation-demodulation system by using harmonic balance method. In *Proceedings of* 13^{th} *European Conference on Circuit Theory and Design (ECCDT'97)*, pages 139–143, Budapest, Hugary, 1997.

[227] F. A. Savacı, M. E. Yalçın, and C. Güzeliş. Steady-state analysis of nonlinearly coupled Chua's circuits with periodic input. *Int. J. Bifurcation and Chaos*, 13(11):3395–3407, 2003.

[228] RSA Data Security. Hardware-based random number generation. Technical report, An RSA Data Security White Paper, 1999. Available for free download :`http://www.intel.com/design/security/rng/Intel_RNG_v2.htm`.

[229] RSA Data Security. Using RSA BSAFE crypto-c with intel random number generator. Technical report, An RSA Data Security White Paper, 1999. Available for free download :`http://www.intel.com/design/security/rng/RSA_BSAFE.htm`.

[230] R. Senani and S. S. Gupta. Implementation of Chua's chaotic circuits using current feedback opamps. *Elecronics Letters*, 34(9):829–830, 1998.

[231] N. S. Sengör, M. E. Yalçın, Y. Çakir M. Üçer, C. Güzeliş, F. Pekergin, and Ö. Morgül. An application of cellular neural network: Maximum clique problem. In *Proceedings of IEEE Int. Workshop on Cellular Neural Networks and Their Applications*, pages 208–211, London, England, 1998.

[232] N. S. Sengör, M. E. Yalçın, Y. Çakir M. Üçer, C. Güzeliş, F. Pekergin, and Ö. Morgül. Solving maximum clique problem by cellular neural network. *Electronics Letters*, 15:1504–1506, 1998.

[233] A. Shamir. On the generation of cryptographically strong pseudorandom sequences. *ACM Transactions on Computer systems*, 1(1):38–44, 1983.

[234] L. P. Shil'nikov. A case of the existence of a denumerable set of periodic motions. *Sov. Math. Dokl.*, 6:163–166, 1965.

[235] C. P. Silva. The double hook attractor in Chua's circuit: Some analytical results. In R. N. Madan, editor, *Chua's circuit: A Paradigm for Chaos*, volume 1 of *World Scientific Series on Nonlinear Science, Series B*, pages 671–710. World Scientific, 1993.

[236] C.P. Silva. Shil'nikov's theorem-a tutorial. *IEEE Trans. Circuits and Systems-I*, 40(10):675–682, 1993.

[237] T. Stojanovski and L. Kocarev. Chaos-based random number generators-part I: Analysis. *IEEE Trans. Circuits and Systems-I*, 48(3):281–288, 2001.

[238] S. H. Strogatz. *Nonlinear Dynamics and Chaos.* Addison Wesley, 1996.

[239] M.A. Styblinski and T.-S. Tang. Experiments in nonconvex optimization: stochastic approximation with function smoothing and simulated annealing. *Neural Networks*, 3:467–483, 1990.

[240] J. A. K Suykens, P. F Curran, and L. O. Chua. Master-slave synchronization using dynamic output feedback. *Int. J. Bifurcation and Chaos*, 7(3):671–679, 1997.

[241] J. A. K Suykens, P. F Curran, and L. O. Chua. Robust synthesis for master-slave synchronization of Lur'e systems. *IEEE Trans. Circuits and Systems-I*, 46(7):841–850, 1999.

[242] J. A. K. Suykens, P. F. Curran, J. Vandewalle, and L. O. Chua. Robust nonlinear H_∞ synchronization of chaotic Lur'e systems. *IEEE Trans. Circuits and Systems-I*, 44(10):891–904, 1997.

[243] J. A. K. Suykens, P. F. Curran, T. Yang, J. Vandewalle, and L. O. Chua. Nonlinear H_∞ synchronization of Lur'e systems : dynamic output feedback case. *IEEE Trans. Circuits and Systems-I*, 44(11):1089–1092, 1997.

[244] J. A. K. Suykens, A. Huang, and L. O. Chua. A family of n-scroll attractors from a generalized Chua's circuit. *Archiv für Elektronik und Ubertragungstechnik*, 51(3):131–138, 1997.

[245] J. A. K. Suykens and J. Vandewalle. Quasilinear approach to nonlinear systems and the design of n-double scroll (n =1,2,3,4,...). *IEE Proceedings-G*, 138(5):595–603, 1991.

[246] J. A. K. Suykens and J. Vandewalle. Generation of n-double scrolls (n= 1,2,3,4,...). *IEEE Trans. Circuits and Systems-I*, 40:861–867, 1993.

[247] J. A. K Suykens and J. Vandewalle. Master-slave synchronization of Lur'e systems. *Int. J. Bifurcation and Chaos*, 7(3):665–669, 1997.

[248] J. A. K. Suykens and J. Vandewalle. The K.U. Leuven time series prediction competition. In J. A. K. Suykens and J. Vandewalle, editors, *Nonlinear modeling : Advanced black-Box Techniques*, pages 241–253, 1998.

[249] J. A. K. Suykens and J. Vandewalle. Chaos synchronization: A lagrange programming network approach. *Int. J. Bifurcation and Chaos*, 10(4):797–810, 2000.

[250] J. A. K Suykens, J. Vandewalle, and L. O. Chua. Nonlinear H_∞ synchronization of chaotic Lur'e systems. *Int. J. Bifurcation and Chaos*, 7(6):1323–1335, 1997.

[251] J. A. K. Suykens, J. Vandewalle, and B. L. R. De Moor. *Artificial Neural Networks for Modelling and Control of Non-Linear systems.* Kluwer Academic Publishers, Boston, 1996.

[252] J. A. K. Suykens, M. E. Yalçın, and J. Vandewalle. A generic class of chaotic and hyperchaotic circuits with synchronization methods. In G. Chen G. and T. Ueta, editors, *Chaos in Circuits and Systems*, volume 11 of *World Scientific Series on Nonlinear Science, Series B*, chapter 8, pages 151–170. World Scientific, 2002.

[253] J. A. K. Suykens, M. E. Yalçın, and J. Vandewalle. Chaotic systems synchronization. In G. Chen and X. Yu, editors, *Chaos Control : Theory and Applications*, volume 292 of *Lecture Notes in Control and Information Sciences*, pages 117–136. Springer-Verlag, 2003.

[254] J. A. K. Suykens, T. Yang, and L. O. Chua. Impulsive synchronization of chaotic Lur'e systems by measurement feedback. *Int. J. Bifurcation and Chaos*, 8(6):1371–1381, 1998.

[255] J. A. K. Suykens, T. Yang, J. Vandewalle, and L. O. Chua. Impulsive synchronization of Lur'e systems: state feedback case. In *Proceedings of the IEEE Conference on Decision and Control (CDC'98)*, pages 1963–1966, Tampa, Florida, 1998.

[256] J. A. K. Suykens, T. Yang, J. Vandewalle, and L. O. Chua. Impulsive control and synchronization of chaos. In G. Chen, editor, *Controlling Chaos and Bifurcations in Engineering Systems*, chapter 13, pages 275–298. CRC Press, Boca Raton, USA, 1999.

[257] J.A.K. Suykens and L. O. Chua. $n-$double scroll hypercubes in $1D-$CNNs. *Int. J. Bifurcation and Chaos*, 7(8):1873–1885, 1997.

[258] J.A.K. Suykens, J. Vandewalle, and B. De Moor. Intelligence and cooperative search by coupled local minimizers. *Int. J. Bifurcation and Chaos*, 11(8):2133–2144, 2001.

[259] J.A.K. Suykens, M.E. Yalçın, and J. Vandewalle. Coupled chaotic simulated annealing processes. In *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS'03)*, volume III, pages 582–585, Bangkok, Thailand, 2003.

[260] I. Szatmari, P. Foldesy, C. Rekeczky, and A. Zarandy. Image processing library for the Aladdin visual computer. In *Proceedings of IEEE Int. Workshop on Cellular Neural Networks and Their Applications*, pages 563–570, Frankfurt, Germany, 2002.

[261] K. S. Tang, K. F. Man, G. Q. Zhong, and G. Chen. Some new circuit design for chaos generation. In G. Chen G. and T. Ueta, editors, *Chaos in Circuits and Systems*, volume 11 of *World Scientific Series on Nonlinear Science, Series B*, chapter 8, pages 171–190. World Scientific, 2002.

[262] K. S. Tang, G. Q. Zhong, G. Chen, and K. F. Man. Generation of n-scroll attractors via sine function. *IEEE Trans. Circuits and Systems-I*, 48(11):1369–1372, 2001.

[263] P. Thiran. Influence of boundary conditions on the behavior of cellular neural networks. *IEEE Trans. Circuits and Systems-I*, 40(3):207–212, 1993.

[264] TISEAN. *Nonlinear Time Series Analysis.* `http://www.mpipks-dresden.mpg.de\~tisean/TISEAN2.1/index.html`.

[265] E. Tissir and A. Hmamed. Further result on stability of $\dot{x}(t) = Ax(t) + Bx(t-\tau)$. *Automatica*, 32:1723–1726, 1996.

[266] M. Tomassini, M. Sipper, and M. Perrenoud. On the generation of high-quality random numbers by two-dimensional cellular automata. *IEEE Trans. on Computers*, 49:1146–1151, 2000.

[267] C. Toumazou and F. J. Lidgey. Current feedback op-amps: A blessing in disguise? *IEEE Circuits and Devices Magazine*, 10(1):43–47, 1994.

[268] C. Toumazou, F. J. Lidgey, and D. Q. Haigh. *Analog IC design: The current mode approach.* London: Peter Peregrinus, 1990.

[269] D. Tschumperle. *PDE-Based Regularization of Multivalued Images and Applications.* PhD thesis, University of Nice-Sophia Antipolis, `ftp://ftp-sop.inria.fr/odyssee/Publications/PhDs/tschumperle:02.pdf`, 2002.

[270] A. van der Schaft. *L_2-gain and passivity techniques in nonlinear control*, volume 218 of *Lecture Notes in Control and Information Sciences.* Springer-Verlag, New York, 1996.

[271] R. Van Schyndel, A. Tirkel, and C. Osborne. A digital watermark. In *Proceedings of the IEEE Int. Conf. on Image Processing*, volume 2, pages 86–90, Austin, Texas, 1994.

[272] M. A. van Wyk and W-H Steeb. Chaos in one-dimensional map. In G. Chen G. and T. Ueta, editors, *Chaos in Circuits and Systems*, volume 11 of *World Scientific Series on Nonlinear Science, Series B*, chapter 29, pages 611–638. World Scientific, 2002.

[273] P. Varona, J. J. Torres, H. D. I. Abarbanel, M. I. Rabinovich, and R. C. Elson. Dynamics of two electrically coupled chaotic neurons: Experimental observations and model analysis. *Biological Cybernetics*, 84:91–102, 2001.

[274] P. L. Venetianer, F. Werblin, T. Roska, and L. O. Chua. Analogic CNN algorithms for some image compression and restoration tasks. *IEEE Trans. Circuits and Systems-I*, 42:278–284, 1995.

[275] M. Vidyasagar. *Nonlinear Systems Analysis.* Prentice-Hall, 1993.

[276] B. Vizvari and G. Kolumban. Quality evaluation of random numbers generated by chaotic sampling phase-locked loops. *IEEE Trans. Circuits and Systems-I*, 45(3):216–224, 1998.

[277] J. Von Neumann. Various techniques used in connection with random digits. *Applied Math Series., Notes by G.E. Forsythe, In National Bureau of Standards*, 12:36–38, 1951.

[278] S. Walton. Information authentication for slippery new age. *Dr. Dobbs Journal*, 20(4):18–26, 1995.

[279] L. Wang and K. Smith. On chaotic simulated annealing. *IEEE Trans. Neural Networks*, 9(4):716–718, 1998.

[280] D. J. Watts and S.H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393:440–442, 1998.

[281] J. C. Willems. Dissipative dynamical systems I: General theory. II: Linear systems with quadratic supply rates. *Archive for Rational Mechanics and Analysis*, 1972.

[282] R. Wolfgang and E. Delp. A watermark for digital images. In *Proceeding of the IEEE Int. Conf. on Image Processing*, pages 219–222, Lausanne, Switzerland, 1996.

[283] R. Wolfgang and E. Delp. Fragile watermarking using VW2D watermark. In *Proceedings of the IS&T/SPIE Conf. on the security and watermarking of multimedia Contents*, pages 204–213, San Jose, California, USA, 1999.

[284] R. B. Wolfgang, C. I. Podilchuk, and E. J. Delp. Perceptual watermarks for digital image and video. *Proceedings of IEEE*, 87(7):1108–1126, 1999.

[285] S. Wolfram. *Cellular Automata and Complexity.* Addison-Wesley Publishing Company, 1994.

[286] C. W. Wu and L. O. Chua. A unified framework for synchronization and control of dynamical systems. *Int. J. Bifurcation and Chaos*, 4:979–989, 1994.

[287] C. W. Wu and L. O. Chua. On the generality of the unfolded Chua's circuit. *Int. J. Bifurcation and Chaos*, 6(5):801–832, 1996.

[288] C. W. Wu and N. F. Rul'kov. Studying chaos via 1-D maps-a tutorial. *IEEE Trans. Circuits and Systems-I*, 40(10):707–721, 1993.

[289] M. Wu and M. Liu. Watermarking for image authentication. In *Proceedings of the IEEE Int. Conf. on Image Processing*, volume 2, pages 437–441, Chicago, Illinois, USA, 1998.

[290] J. L. Wyatt, L. O. Chua, J. W. Gannett, I. C. Göknar, and D. N. Green. Energy concepts in state-space theory of nonlinear n-ports: Part-I-Passivity. *IEEE Trans. Circuits and Systems*, 28(1):48–61, 1981.

[291] S. Xavier de Souza, M. E. Yalçın, J. A. K. Suykens, and J. Vandewalle. Automatic chip-specific CNN template optimization using adaptive simulated annealing. In *Proceedings of the 16th European Conference on Circuits Theory and Design (ECCTD'03)*, 2003.

[292] S. Xavier de Souza, M. E. Yalçın, J. A. K. Suykens, and J. Vandewalle. Toward CNN chip-specific robustness. *IEEE Trans. Circuits and Systems-I*, 51(5):892–902, 2004.

[293] L. Xie and G. Arce. Joint wavelet compression and authentication watermarking. In *Proc. of the IEEE Int. Conf. on Image Processing*, volume 2, pages 427–431, Chicago, Illinois, USA, 1998.

[294] M. E. Yalçın and C. Güzeliş. CNN's with radial basis input function. In *Proceedings of IEEE Int. Workshop on Cellular Neural Networks and Their Applications*, pages 231–236, Sevilla, Spain, 1996.

[295] M. E. Yalçın, B. Preneel, J. Vandewalle, and M. Csapodi. Algorithms for image authenticity and integrity verification using hardware platform and control software. Technical Report 99-94, K.U. Leuven, ESAT-SISTA, 1999.

[296] M. E. Yalçın and F. A. Savacı. Realization of Chua's circuit by using voltage controlled voltage source (VCVS) and its harmonic balance analysis. *ARI (ITU-Springer Verlag)*, 51(3):169–174, 1999.

[297] M. E. Yalçın, F. A. Savacı, and C. Güzeliş. A chaotic state modulation-demodulation system using nonlinearly coupled Chua's circuits. In *Proceedings of the The 5^{th} International Specialist Workshop on Nonlinear Dynamics of Electronic Systems (NDES-97)*, pages 197–202, Moskow, Russia, 1997.

[298] M. E. Yalçın, L. Şavkay, and F. A. Savacı. New realization of Chua's circuit by using voltage controlled voltage source (VCVS). In *Proceedings of Int. Workshop on Nonlinear Dynamics of Electronic Systems (NDES'96)*, pages 259–264, Sevilla, Spain, 1996.

[299] M. E. Yalçın, J. A. K. Suykens, and J. Vandewalle. Experimental confirmation of a nonlinear H_∞ synchronizationscheme with 5-scroll attractors. In *Proceedings of the International Workshop on nonlinear dynamics of electronic systems (NDES'99)*, pages 257–260, Ronne, Denmark, 1999.

[300] M. E. Yalçın, J. A. K. Suykens, and J. Vandewalle. Experimental confirmation of nonlinear H_∞ synchronization for Chua's circuit. In *Proceedings of the European Conferance on Circuit Theory and Design (ECCTD'99)*, pages 177–180, Stresa, Italy, 1999.

[301] M. E. Yalçın, J. A. K. Suykens, and J. Vandewalle. On the realization of n-scroll attractors. In *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS'99)*, volume 5, pages 483–486, Orlando, Florida, USA, 1999.

[302] M. E. Yalçın, J. A. K. Suykens, and J. Vandewalle. Experimental confirmation of 3- and 5-scroll attractors from a generalized Chua's circuit. *IEEE Trans. Circuits and Systems-I*, 47(3):425–429, 2000.

[303] M. E. Yalçın, J. A. K. Suykens, and J. Vandewalle. Hyperchaotic n-scroll attractors. In *Proceedings of the 8^{th} Int. Workshop on Nonlinear Dynamics of Electronic Systems (NDES 2000)*, pages 25–28, Catania, Italy, 2000.

[304] M. E. Yalçın, J. A. K. Suykens, and J. Vandewalle. Mutual synchronization of Lur'e systems with time delay. In *Proceedings of the 2000 International Symposium on Nonlinear Theory and its Applications (NOLTA 2000)*, pages 241–244, Dresden, Germany, 2000.

[305] M. E. Yalçın, J. A. K. Suykens, and J. Vandewalle. Master - slave synchronization of Lur'e systems with time-delay. *Int. J. Bifurcation and Chaos*, 11(6):1707–1721, 2001.

[306] M. E. Yalçın, J. A. K. Suykens, and J. Vandewalle. A double scroll based true random bit generator. In *Proceedings of IEEE Int. Symp. on Circuits and Systems (ISCAS 2004)*, 2004. In press.

[307] M. E. Yalçın, J. A. K. Suykens, and J. Vandewalle. True random bit generation from a double scroll attractor. *IEEE Trans. Circuits and Systems-I*, 2004. In press.

[308] M. E. Yalçın and J. Vandewalle. Fragile watermarking and unkeyed hash function implementation for image authentication on CNN-UM. In *Proceedings of IEEE Int. Workshop on Cellular Neural Networks and Their Applications*, pages 399–406, Frankfurt, Germany, 2002.

[309] M. E. Yalçın, J. Vandewalle, P. Arena, A. Basile, and L. Fortuna. Watermarking on CNN-UM for image and video authentication. *Int. J. Circuit Theory and Applications*, 2004. Accepted.

[310] M.E. Yalçın, S. Ozoğuz, J.A.K. Suykens, and J. Vandewalle. $2D$-grid scroll attractors. In *Proccedings of the 9^{th} Workshop on Nonlinear Dynamics of Electronic Systems (NDES 2001)*, pages 181–184, Delft, The Nederlands, 2001.

[311] M.E. Yalçın, S. Ozoğuz, J.A.K. Suykens, and J. Vandewalle. $3D$-grid scroll attractors. In *Proccedings of the* 15^{th} *European Conference on Circuit Theory and Design (ECCTD'01)*, volume III, pages 425–428, Helsinki, Finland, 2001.

[312] M.E. Yalçın, S. Ozoğuz, J.A.K. Suykens, and J. Vandewalle. Families of scroll grid attractors. *Int. J. Bifurcation and Chaos*, 12(1):23–41, 2001.

[313] M.E. Yalçın, S. Ozoğuz, J.A.K. Suykens, and J. Vandewalle. $n-$scroll chaos generators: a simple circuit model. *Electronics Letters*, 37(3):147–148, 2001.

[314] M.E. Yalçın, S. Ozoğuz, J.A.K. Suykens, and J. Vandewalle. Scroll maps from n-scroll attractors. In *Proccedings of the* 10^{th} *Workshop on Nonlinear Dynamics of Electronic Systems (NDES 2002)*, pages 45–48, Izmir, Turkey, 2002.

[315] T. Yang and L. O. Chua. Impulsive stabilization for control and synchronization of chaotic systems: theory and application to secure communication. *IEEE Trans. Circuits and Systems-I*, 44(10):976–988, 1997.

[316] T. Yang and L. O. Chua. Applications of chaotic digital code-division multiple access (CDMA) to cable communication. *Int. J. Bifurcation and Chaos*, 8(8):1657–1669, 1998.

[317] X. S. Yang and Q. Li. Generate n-scroll attractor in linear system by scalar output feedback. *Chaos, Solitons & Fractals*, 18(1):25–29, 2003.

[318] M. Yeung and F. Mintez. Invisible watermarking for image verification. *Journal of Electronic Imaging*, 7(3):578–591, 1988.

[319] A. Zarandy. The art of CNN template design. *Int. J. Circuit Theory and Applications*, 27(1):5–23, 1999.

[320] A. Zarandy, C. Rekeczky, P. Foldesy, and I. Szatmari. The new framework of applications - the aladdin system. *Journal of Circuits, Systems, and Computers*, 12(6):769–781, 2003.

[321] A. Zarandy, A. Stoffels, T. Roska, and L. O. Chua. Implementation of binary and gray-scale mathematical morphology on the CNN universal machine. *IEEE Trans. Circuits and Systems-I*, 50(7):163–168, 1998.

[322] S. Zhang and A.G. Constantinides. Lagrange programming neural networks. *IEEE Trans. Circuits and Systems-II*, 39(7):441–452, 1992.

[323] G. Q. Zhong and F. Ayron. Experimental confirmation of chaos from Chua's circuit. *Int. J. Circuit Theory and Applications*, 13(1):93–98, 1985.

[324] G. Q. Zhong and F. Ayron. Periodicity and chaos in Chua's circuit. *IEEE Trans. Circuits and Systems*, 32:501–503, 1985.

[325] G. Q. Zhong, K. F. Man, and G. R. Chen. A systematic approach to generating n-scroll attractors. *Int. J. Bifurcation and Chaos*, 12(12):2907–2915, 2002.

[326] F. Zou and J. A. Nossek. Bifurcation and chaos in CNN. *IEEE Trans. Circuits and Systems-I*, 40(3):166–173, 1993.

[327] J. M. Zurada. *Introduction to Artificial Neural Systems*. PWS Publishing Company, 1992.