Springer Complexity

Springer Complexity is an interdisciplinary program publishing the best research and academic-level teaching on both fundamental and applied aspects of complex systems – cutting across all traditional disciplines of the natural and life sciences, engineering, economics, medicine, neuroscience, social and computer science.

Complex Systems are systems that comprise many interacting parts with the ability to generate a new quality of macroscopic collective behavior the manifestations of which are the spontaneous formation of distinctive temporal, spatial or functional structures. Models of such systems can be successfully mapped onto quite diverse "real-life" situations like the climate, the coherent emission of light from lasers, chemical reaction-diffusion systems, biological cellular networks, the dynamics of stock markets and of the internet, earthquake statistics and prediction, freeway traffic, the human brain, or the formation of opinions in social systems, to name just some of the popular applications.

Although their scope and methodologies overlap somewhat, one can distinguish the following main concepts and tools: self-organization, nonlinear dynamics, synergetics, turbulence, dynamical systems, catastrophes, instabilities, stochastic processes, chaos, graphs and networks, cellular automata, adaptive systems, genetic algorithms and computational intelligence.

The three major book publication platforms of the Springer Complexity program are the monograph series "Understanding Complex Systems" focusing on the various applications of complexity, the "Springer Series in Synergetics", which is devoted to the quantitative theoretical and methodological foundations, and the "SpringerBriefs in Complexity" which are concise and topical working reports, case-studies, surveys, essays and lecture notes of relevance to the field.

In addition to the books in these two core series, the program also incorporates individual titles ranging from textbooks to major reference works.

Understanding Complex Systems

Founding Editor: J.A. Scott Kelso

Future scientific and technological developments in many fields will necessarily depend upon coming to grips with complex systems. Such systems are complex in both their composition – typically many different kinds of components interacting simultaneously and nonlinearly with each other and their environments on multiple levels – and in the rich diversity of behavior of which they are capable.

The Springer Series in Understanding Complex Systems series (UCS) promotes new strategies and paradigms for understanding and realizing applications of complex systems research in a wide variety of fields and endeavors. UCS is explicitly transdisciplinary. It has three main goals: First, to elaborate the concepts, methods and tools of complex systems at all levels of description and in all scientific fields, especially newly emerging areas within the life, social, behavioral, economic, neuro- and cognitive sciences (and derivatives thereof); second, to encourage novel applications of these ideas in various fields of engineering and computation such as robotics, nano-technology and informatics; third, to provide a single forum within which commonalities and differences in the workings of complex systems may be discerned, hence leading to deeper insight and understanding.

UCS will publish monographs, lecture notes and selected edited contributions aimed at communicating new findings to a large multidisciplinary audience.

Santo Banerjee • Lamberto Rondoni • Mala Mitra
Editors

Applications of Chaos and Nonlinear Dynamics in Science and Engineering – Vol. 2

Springer

Editors
Santo Banerjee
Institute for Mathematical Research
University Putra Malaysia
Malaysia

and

Department of Complexity
and Network Dynamics
International Science Association
Ankara
Turkey

Lamberto Rondoni
Dipartimento di Matematica
Politecnico di Torino
Torino
Italy

and

INFN, Sezione di Torino
Via P. Giuria 1
10125 Torino
Italy

Mala Mitra
Department of Physics
Camellia School of Engineering
and Technology
Kolkata
India

ISSN 1860-0832
ISBN 978-3-642-29328-3
DOI 10.1007/978-3-642-29329-0
Springer Heidelberg New York Dordrecht London

ISSN 1860-0840 (electronic)
ISBN 978-3-642-29329-0 (eBook)

Library of Congress Control Number: 2011936888

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

SB: To my son, Sumangal
LR: To my wife, Gabriella
MM: To my parents, Sujata and Sukumar Mitra

Preface

This is the second volume of the collection of essays entitled *Applications of Chaos and Nonlinear Dynamics in Engineering*, meant to introduce the interested readers to the subject of the ever-growing field of engineering applications of the modern theory of dynamical systems. This purpose has been pursued by means of essays written in the form of tutorials. The 10 chapters published in the first volume were organized in five parts, each concerning one field of present-day engineering:

I. Nonlinearity and Computer Simulations
II. Chaos and Nonlinear Dynamics in Electrical Engineering
III. Chaos and Nonlinear Dynamics in Building Mechanism and Fluid Dynamics
IV. Chaos in Robotics
V. Chaos and Nonlinear Dynamics in Communication

The present volume also contains ten chapters, subdivided into the following five parts:

I. Nonlinearity in Control Systems and Geo Engineering
II. Nonlinear Dynamics and Chaos in Electronics
III. Nonlinear Dynamics in Stochastic Networks
IV. Nonlinear Dynamics in Transport and Mechanical Engineering
V. Chaos Theory in Communication and Cryptography

The first contribution, by Bingo Wing-Kuen Ling, consists of a numerical study of the dynamics of HIV, which provides indications on the effectiveness of therapies, for medical doctors to consider. The second contribution, by Devanjan Bhattacharya, J.K. Ghosh, and Swej Kumar Sharma, gives an application of fuzzy theory to the development of landslides. The third contribution, by Hamid Reza Karimi, concerns the *nonlinear control* method known as sliding mode control, with applications in chaotic dynamics. The fourth chapter, by M.P. Hanias, H.E. Nistazakis, and G.S. Tombras, addresses the classical and technological important issue of transistor circuits, which may be driven into chaotic states by external

perturbations. The fifth essay, Nicole Abaid and Maurizio Porfiri, deals with numerosity-constrained networks and their applications to consensus problems and to synchronization.

The sixth chapter, by Gianpiero Mastinu, Fabio Della Rossa, and Carlo Piccardi, investigates the stability of automobiles travelling on straight or curved roads. The seventh article, by Shane D. Ross and Phanindra Tallapragada, discusses the features of chaotic phase space transport in finite time, nonautonomous systems, which are of interest in many different situations, including the study of geophysical flows and mixing in microchannels. The eighth chapter, by Congxu Zhu and Kehui Sun, considers the problem of copyrights and digital watermarking, and explains how chaotic dynamics can be used for copyright protection. The ninth chapter, by M.R.K. Ariffin, N.M.G. Al-Saidi, M.R.M. Said, Z. Mahad, and M.A. Daud, concerns the use of chaos-based techniques for cryptosystems. The tenth chapter, by S. Banerjee, considers synchronization and introduces a robust method of digital cryptography based on genetic algorithms.

Like the chapters published in volume 1, we hope that also this second collection of chapter will be found useful by professionals as well as undergraduate and graduate students of applied sciences.

We wish to express our gratitude to the staff of Springer Verlag, for their invaluable help and support throughout this work. In particular, we would like to thank Dr. Christian Caron (editor, Springer Physics), Gabriele Hakuba (editorial assistant), Benjamin Feuchter, and Anitha Murugaiyan.

Torino, 4 November 2011

S. Banerjee, L. Rondoni, and M. Mitra

Contents

Part 5 Chaos Theory in Communication and Cryptography

Part 1
Nonlinearity in Control Systems and Geo Engineering

Chapter 1
HIV Therapy via Noncomputational Optimal Control Approach

Bingo Wing-Kuen Ling

1.1 Introduction

In 2005, there were about 38.6 millions of adults and children suffering from the human immunodeficiency virus (HIV). The total infected population is increasing year by year [1]. There are about 20 medications approved by the Food and Drug Administration (FDA) of the US government. These medications can suppress the acquired immune deficiency syndrome (AIDS).

It was reported in [2–20] that the response of the CD4 lymphocyte population, the CD8 lymphocyte population and the HIV-1 viral load can be characterized by a first order nonlinear differential equation. In this chapter, we numerically show that the system dynamics is sensitive to both the initial condition and the system parameters. This implies that the system is chaotic and exhibits a bifurcation behavior. Moreover, the concentrations of the CD4 lymphocyte population, the CD8 lymphocyte population, the HIV-1 viral load and the dose are required to be non-negative and bounded, this implies that the positivity constraint of the system states and that of the dose concentration is required to be satisfied [21]. Because of the chaotic nature of the system and the positivity constraint, the system is very difficult to be controlled. Furthermore, the system has two equilibrium points and one of them is unstable [22]. This implies that a very small derivation of the system vectors from the equilibrium point would cause the system trajectory to diverge locally. In order to control the concentration of the HIV-1 viral load being exactly equal to zero within a certain period of time, zero boundary conditions of the controlled system are required to be satisfied. However, it was reported in [23] that the corresponding linearized system (linearized at the unstable equilibrium point) is uncontrollable, so linear control methods would usually result to the loss of control near the equilibrium point.

B.W.-K. Ling (✉)
School of Engineering, University of Lincoln, Lincoln, Lincolnshire, LN6 7TS, UK
e-mail: wling@lincoln.ac.uk

S. Banerjee et al. (eds.), *Applications of Chaos and Nonlinear Dynamics in Science and Engineering - Vol. 2*, Understanding Complex Systems,
DOI 10.1007/978-3-642-29329-0_1,

The simplest existing strategy for controlling the HIV-1 viral load is via the P control law [22, 24, 25], that is, the dose concentration is directly proportional to the concentration of the HIV-1 viral load. However, this control strategy fails to satisfy the positivity constraint and in general this control method does not guarantee the asymptotical stability of the system. Optimal control methods [26, 27] are also proposed. However, all these control methods are derived based on the corresponding linearized model, in which the system will lose control when the trajectory is close to the equilibrium point [23].

As the medications have strong side effects, medical professions would prefer to initiate the therapy only at certain situations [23]. However, there is no standard rule for the determination of the time instant that the therapy initiates. Also, the corresponding dose concentration is unknown. In this chapter, we propose to initiate the therapy based on both the concentration of the HIV-1 viral load and the ratio of the CD4 lymphocyte population to the CD8 lymphocyte population. If the concentration of the HIV-1 viral load is higher than a threshold, then the first type of therapy will be applied. If the concentration of the HIV-1 viral load is lower than or equal to the threshold and the ratio of the CD4 lymphocyte population to the CD8 lymphocyte population is greater than another threshold, then the second type of therapy will be applied. Otherwise, no therapy will be applied. The advantages of the proposed control strategy are that the therapy can be stopped under certain conditions, while the state variables of the overall system is asymptotically stable with fast convergent rate, the concentration of the controlled HIV-1 viral load is monotonic decreasing, as well as the positivity constraint of the system states and that of the dose concentration is guaranteed to be satisfied.

The outline of this chapter is as follows: The HIV model is reviewed in Sect. 1.2. In Sect. 1.3, the sensitivity of the initial condition and the system parameters against the system dynamics is simulated. The obtained results imply that the system is chaotic and exhibits a bifurcation behavior. In Sect. 1.4, a control strategy is proposed. Computer numerical simulation results are also presented. Finally, a conclusion is drawn in Sect. 1.5.

1.2 HIV Model

Denote a, b, c, d, e and f as the system parameters, in which all are positive. Denote $x(t)$, $y(t)$ and $z(t)$ as the concentrations of the CD4 lymphocyte population, the CD8 lymphocyte population and the HIV-1 viral load, respectively. Denote x_0 and y_0 as the normal unperturbed concentrations of the CD4 and the CD8 lymphocyte population, respectively, in which they are also positive. The response of the concentrations of the CD4 lymphocyte population, the CD8 lymphocyte population and the HIV-1 viral load can be characterized by the following first order nonlinear differential equation [2–20]:

$$\frac{d}{dt}x(t) = a(x_0 - x(t)) - bx(t)z(t), \tag{1.1a}$$

$$\frac{d}{dt}y(t) = c(y_0 - y(t)) + dy(t)z(t) \tag{1.1b}$$

and

$$\frac{d}{dt}z(t) = z(t)(ex(t) - fy(t)). \tag{1.1c}$$

It can be checked easily that the equilibrium points of the system are

$$[x_0 \quad y_0 \quad 0]^T \tag{1.2a}$$

and

$$\left[\frac{aedx_0 + bfcy_0}{e(ad+bc)} \quad \frac{aedx_0 + bfcy_0}{f(ad+bc)} \quad \frac{ac(ex_0 - fy_0)}{aedx_0 + bfcy_0}\right]^T. \tag{1.2b}$$

1.3 Sensitivity of Initial Condition and System Parameters Against System Dynamics

Figure 1.1a shows the state responses of the system, while Fig. 1.1b shows the corresponding state trajectory, in which $a = 0.25, b = 50, c = 0.25, d = 10, e = 0.01$, $f = 0.006$, $x_0 = 1000$, $y_0 = 550$, $x(0) = x_0$, $y(0) = y_0$ and $z(0) = 0.03$. It can be seen from Fig. 1.1b that the system exhibits a complex behavior. Figure 1.2 plots the Lyapunov exponents of the system at different time instants with the system parameters being the same as that applied in Fig. 1.1 and with the initial condition being the state vectors at the corresponding time instants. It can be seen from Fig. 1.2 that the Lyapunov exponent of $z(t)$ is positive during the transient moment and at $t = 4.15$ year. Figure 1.3 plots the difference of the state responses between the following two sets of initial conditions with the same set of system parameters applied in Fig. 1.1: $x(0) = 442$, $y(0) = 736$ and $z(0) = 0.0063$, and its 1% increment, that is $x(0) = 446.42$, $y(0) = 743.36$ and $z(0) = 0.0064$. It can be seen from Fig. 1.3 that the difference of the state responses between the above two sets of initial conditions are very large even though these two sets of initial conditions are differed only by 1%. This implies that the system is very sensitive to the initial condition. To investigate the sensitivity of the system parameters against the system dynamics, since there are six system parameters in the HIV model, namely, a, b, c, d, e and f, the range of the concentrations of the CD4 lymphocyte population, the CD8 lymphocyte population and the HIV-1 viral load are plotted with respect to these system parameters. Figure 1.4a–c plot the ranges of $x(t)$, $y(t)$ and $z(t)$ as a varies when $b = 50, c = 0.25, d = 10, e = 0.01, f = 0.006, x_0 = 1000, y_0 = 550, x(0) = x_0$,

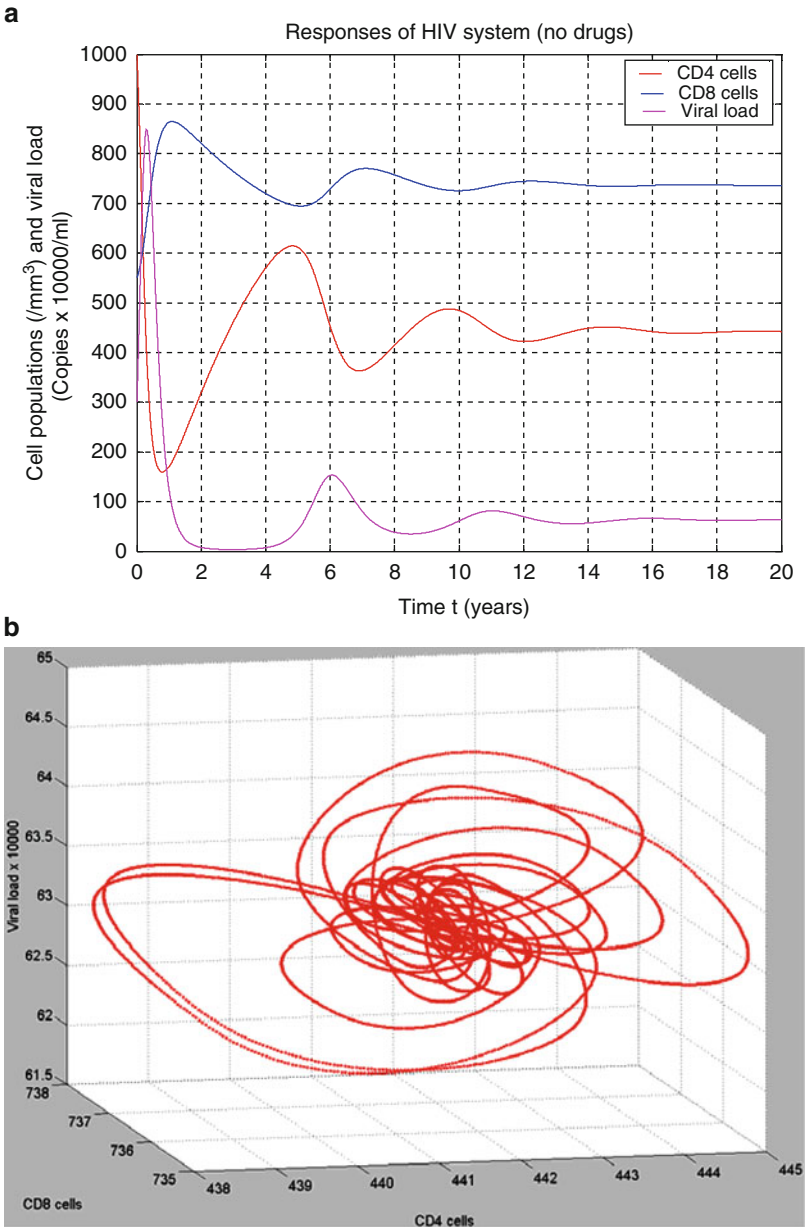

Fig. 1.1 (**a**) State responses of the system. (**b**) The state trajectory of the system

$y(0) = y_0$ and $z(0) = 0.03$. Similarly, Figs. 1.5–1.9 plot the corresponding ranges as the parameter b to the parameter f vary, respectively. From these computer numerical simulation results, we can see that the parameters d and e are the most sensitive to

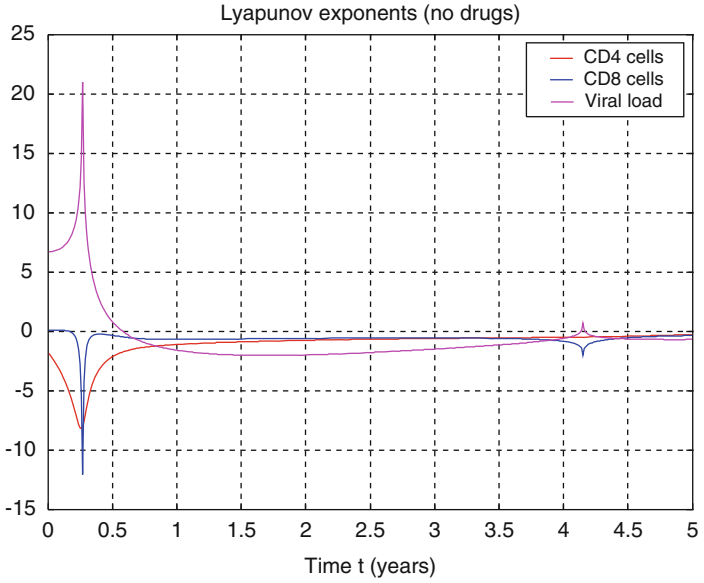

Fig. 1.2 Lyapunov exponents of the system

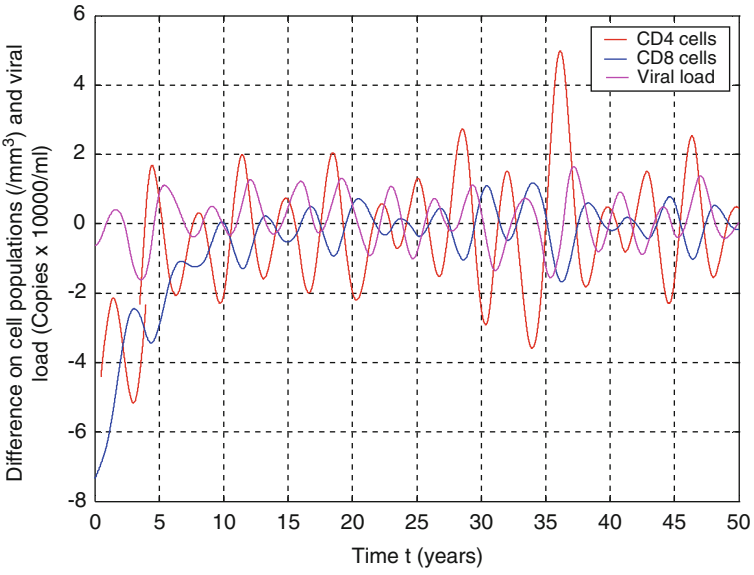

Fig. 1.3 Difference of the responses of the CD4 lymphocyte population, the CD8 lymphocyte population and the HIV-1 viral load between the two different sets of initial conditions

the system dynamics. Hence, we can conclude that the system is chaotic and exhibits a bifurcation phenomenon. This chaotic nature of the system causes the formulation of the control strategy being very challenging.

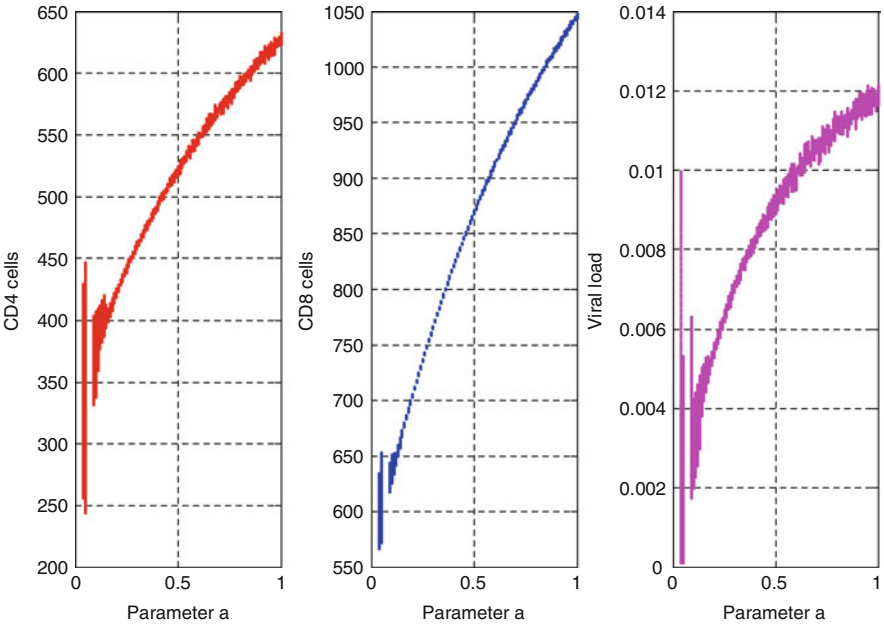

Fig. 1.4 Ranges of the CD4 lymphocyte population, the CD8 lymphocyte population and the HIV-1 viral load as the parameter *a* varies

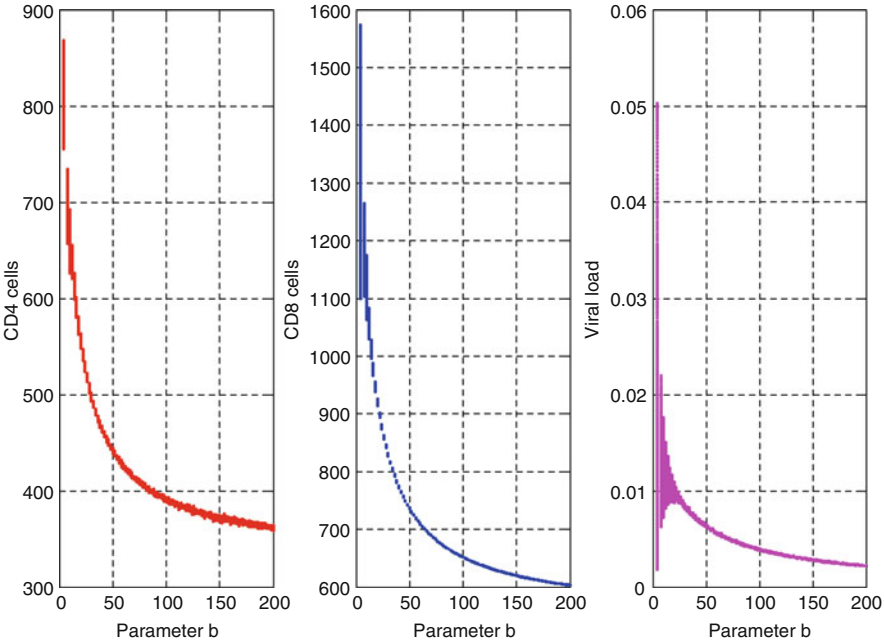

Fig. 1.5 Ranges of the CD4 lymphocyte population, the CD8 lymphocyte population and the HIV-1 viral load as the parameter *b* varies

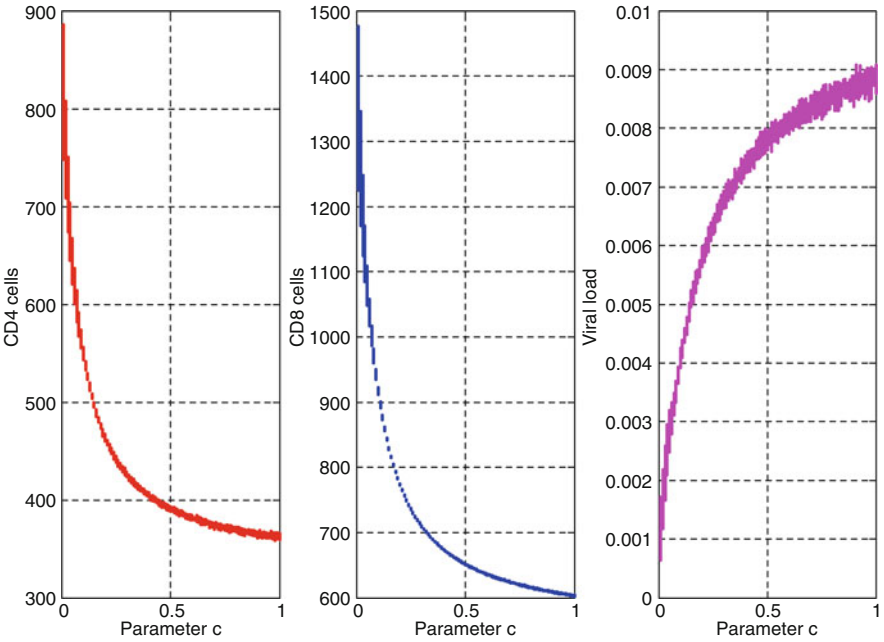

Fig. 1.6 Ranges of the CD4 lymphocyte population, the CD8 lymphocyte population and the HIV-1 viral load as the parameter c varies

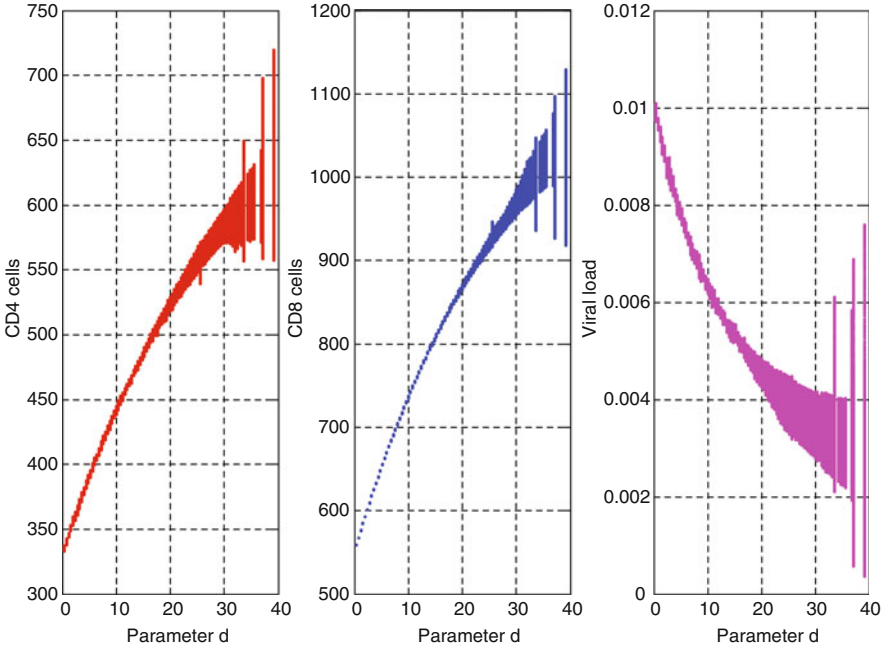

Fig. 1.7 Ranges of the CD4 lymphocyte population, the CD8 lymphocyte population and the HIV-1 viral load as the parameter d varies

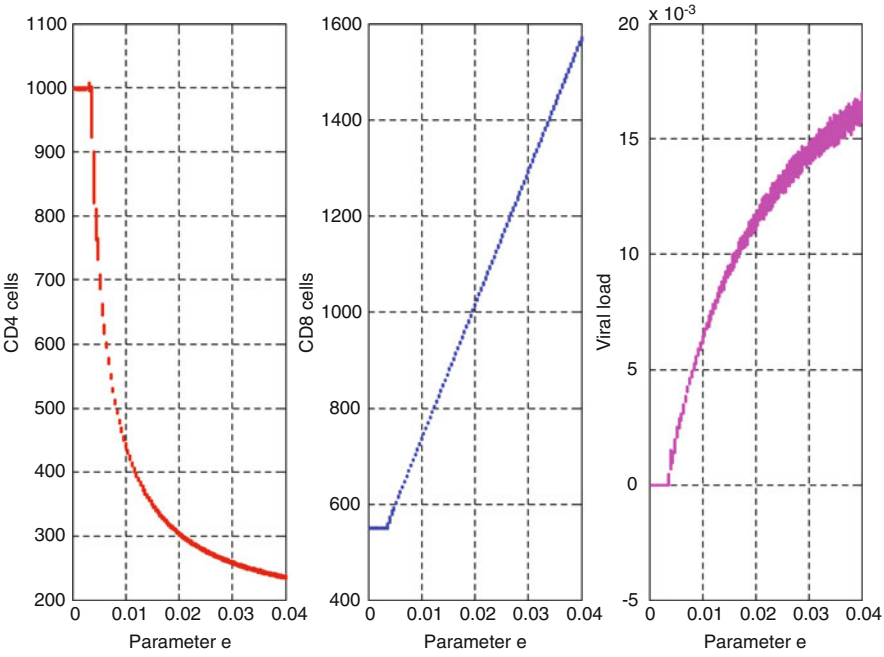

Fig. 1.8 Ranges of the CD4 lymphocyte population, the CD8 lymphocyte population and the HIV-1 viral load as the parameter *e* varies

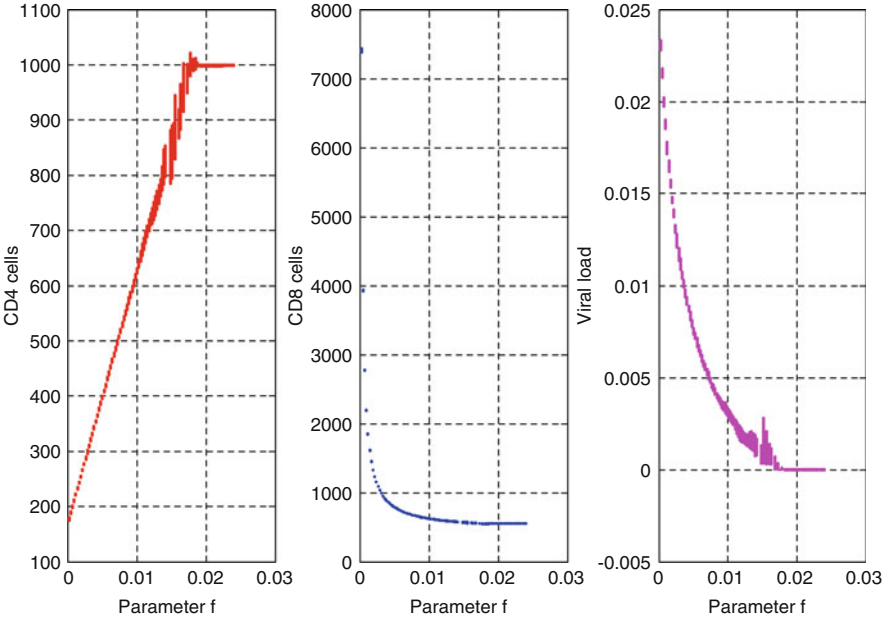

Fig. 1.9 Ranges of the CD4 lymphocyte population, the CD8 lymphocyte population and the HIV-1 viral load as the parameter *f* varies

1.4 Proposed Control Strategy

Denote $U(t)$ as the dose concentration. This corresponds to the control force of the system. Assume that $U(t)$ is applying directly to $z(t)$, that is:

$$\frac{d}{dt}x(t) = a(x_0 - x(t)) - bx(t)z(t), \tag{1.3a}$$

$$\frac{d}{dt}y(t) = c(y_0 - y(t)) + dy(t)z(t) \tag{1.3b}$$

and

$$\frac{d}{dt}z(t) = z(t)(ex(t) - fy(t)) - U(t). \tag{1.3c}$$

Consider the following control force

$$U(t) = \frac{e^2(x(t))^2 z(t)}{4fy(t)} \tag{1.4}$$

and define

$$V(t) = (z(t))^2. \tag{1.5}$$

It can be easily seen that $V(t) > 0$ for $z(t) \neq 0$, and

$$\frac{d}{dt}z(t) = -\frac{z(t)(ex(t) - 2fy(t))^2}{4fy(t)}. \tag{1.6}$$

Consequently, we have

$$\frac{d}{dt}V(t) = -\frac{(z(t))^2(ex(t) - 2fy(t))^2}{2fy(t)}. \tag{1.7}$$

As

$$\frac{d}{dt}x(t) = a(x_0 - x(t)) - bx(t)z(t), \tag{1.8}$$

we have

$$x(t) = \frac{\int_{-\infty}^{t} ax_0 \exp\left(\int_{-\infty}^{\xi} (a + bz(\tau))d\tau\right)d\xi}{\exp\left(\int_{-\infty}^{t} (a + bz(\tau))d\tau\right)}. \tag{1.9}$$

Similarly, as

$$\frac{d}{dt}y(t) = c(y_0 - y(t)) + dy(t)z(t), \tag{1.10}$$

we have

$$y(t) = \frac{\int_{-\infty}^{t} cy_0 \exp\left(\int_{-\infty}^{\xi} (c - dz(\tau))d\tau\right)d\xi}{\exp\left(\int_{-\infty}^{t} (c - dz(\tau))d\tau\right)}. \tag{1.11}$$

Since a, c, x_0 and y_0 are positive, from (1.9) to (1.11), we can conclude that $x(t) > 0$ and $y(t) > 0 \, \forall t \geq 0$. By putting $y(t) > 0$ and $f > 0$ to (1.7), we can conclude that $\frac{d}{dt}V(t) < 0 \forall t \geq 0$. Since we assume that $z(0) > 0$, we have $\frac{d}{dt}z(t) < 0 \forall t \geq 0, z(t) > 0$ $\forall t \geq 0$ and $\lim_{t \to +\infty} z(t) = 0$. By putting $f > 0$, $y(t) > 0$ and $z(t) > 0$ to (1.4), we have $U(t) > 0 \forall t \geq 0$. When $z(t) \approx 0 \forall t \geq T$, where T is a positive real number, from (1.9) to (1.11), we have

$$x(t) \approx x_0 + (x(T) - x_0)\exp(a(T - t)) \tag{1.12}$$

and

$$y(t) \approx y_0 + (y(T) - y_0)\exp(c(T - t)) \quad \forall t \geq T. \tag{1.13}$$

Since a and c are positive, we have

$$\lim_{t \to +\infty} x(t) \approx x_0 \tag{1.14}$$

and

$$\lim_{t \to +\infty} y(t) \approx y_0. \tag{1.15}$$

Consequently, both $x(t)$ and $y(t)$ are bounded and converge to the corresponding equilibrium values. Hence, this control strategy can guarantee the asymptotical stability of the system, the monotonic decrement of the concentration of the HIV-1 viral load, as well as the satisfaction of the positivity constraint of the system states and that of the dose concentration.

Although this control strategy guarantees that the system states will converge to the equilibrium point, the convergent rate is slow because $V(t)$ is independent of both $x(t)$ and $y(t)$. Now, consider another control force as follows:

$$U(t) = \frac{e^2(x(t))^2 z(t)}{4fy(t)} + \frac{k_2 d}{k_3}\left(y(t) - \frac{y_0}{2}\right)^2 + \frac{k_1 b x_0^2}{4k_3}, \tag{1.16}$$

and define

$$V(t) = k_1(x_0 - x(t))^2 + k_2(y_0 - y(t))^2 + k_3(z(t))^2, \tag{1.17}$$

Where k_1 and k_2 are non-negative constants as well as k_3 is a positive constant. Obviously, the previous control strategy is a particular case of this control strategy. Now, we have

$$\frac{d}{dt}z(t) = -\frac{z(t)(ex(t) - 2fy(t))^2}{4fy(t)} - \frac{k_2 d}{k_3}\left(y(t) - \frac{y_0}{2}\right)^2 - \frac{k_1 b x_0^2}{4k_3} \tag{1.18}$$

and

$$\begin{aligned}\frac{d}{dt}V(t) = &-\frac{k_3(z(t))^2(ex(t) - 2fy(t))^2}{2fy(t)} - 2k_2c(y_0 - y(t))^2 - 2k_1a \\ &\times (x_0 - x(t))^2 - 2k_1bz(t)\left(x(t) - \frac{x_0}{2}\right)^2 - \frac{k_2 dz(t)y_0^2}{2}.\end{aligned} \tag{1.19}$$

It can be checked easily that $V(t) > 0$ for $(x(t), y(t), z(t)) \neq (x_0, y_0, 0)$. Similarly, from (1.9) to (1.11), we have $x(t) > 0$ and $y(t) > 0\ \forall t \geq 0$. If $z(t) > 0\ \forall t \geq 0$, then $U(t) > 0$, $\frac{d}{dt}V(t) < 0$ and $\frac{d}{dt}z(t) < 0 \forall t \geq 0$. Hence, the success of this control strategy highly depends whether $z(t) > 0 \quad \forall t \geq 0$ or not. However, in general it is not guaranteed that $z(t) > 0\ \forall t \geq 0$ is satisfied even though this control strategy could achieve a fast convergent rate because $V(t)$ depends on both $x(t)$ and $y(t)$.

Now, consider an uncontrolled case, that is:

$$U(t) = 0. \tag{1.20}$$

Define

$$V(t) = (z(t))^2, \tag{1.21}$$

then we have

$$\frac{d}{dt}z(t) = z(t)(ex(t) - fy(t)) \tag{1.22}$$

and

$$\frac{d}{dt}V(t) = 2(z(t))^2(ex(t) - fy(t)). \tag{1.23}$$

It can be checked easily that $V(t) > 0$ for $z(t) \neq 0$. Similarly, from (1.9) to (1.11), we have $x(t) > 0$ and $y(t) > 0\ \forall t \geq 0$. If $ex(t) - fy(t) < 0$, then $\frac{d}{dt}V(t) < 0$. Since we

assume that $z(0) > 0$, we have $\frac{d}{dt}z(t) < 0\ \forall t \geq 0$, $z(t) > 0\ \forall t \geq 0$ and $\lim_{t\to+\infty} z(t) = 0$. Similarly, we have $\lim_{t\to+\infty} x(t) \approx x_0$ and $\lim_{t\to+\infty} y(t) \approx y_0$. Hence, the success of the uncontrolled strategy highly depends whether $ex(t) - fy(t) < 0$ or not. However, in general it is not guaranteed that $ex(t) - fy(t) < 0$ is satisfied even though the uncontrolled strategy does not require patients taking medications and patients do not need to tolerate the side effects of the medications.

Now, consider the following control strategy:

Theorem 1.1 *Assume that* $a, b, c, d, e, f, x_0, y_0, k_1, k_2, k_3, x(0), y(0), z(0) > 0$. *Denote* α *as a positive constant.*

Case I: If $z(t) > \alpha$, then $U(t) = \frac{e^2(x(t))^2 z(t)}{4fy(t)} + \frac{k_2 d}{k_3}\left(y(t) - \frac{y_0}{2}\right)^2 + \frac{k_1 b x_0^2}{4k_3}$.
Case II: If $\alpha \geq z(t) > 0$ and $ex(t) - fy(t) \geq 0$, then $U(t) = \frac{e^2(x(t))^2 z(t)}{4fy(t)}$.
Case III: If $\alpha \geq z(t) > 0$ and $ex(t) - fy(t) < 0$, then $U(t) = 0$.

The overall controlled system is asymptotically stable, the concentration of the HIV-1 viral load is monotonic decreasing, as well as the positivity constraint of the system states and that of the dose concentration is guaranteed to be satisfied.

Proof. Since $\frac{d}{dt}z(t) < 0$ for all these three cases and $z(0) > 0$, the system can only switch from Case I to either Case II or Case III, but not vice versa. If $z(0) \leq \alpha$, then the system can only operate in Case II and Case III. If $z(0) > \alpha$, as $\frac{d}{dt}z(t) < 0$, $z(t)$ is decreasing until it reaches the value α and then switches to either Case II or Case III. Hence, the objective of the control strategy defined by Case I is to accelerate the transient response. Once the system is operating under the control strategy defined by either Case II or Case III, as $\frac{d}{dt}V(t) < 0$, $\frac{d}{dt}z(t) < 0$, $x(t) > 0$, $y(t) > 0$, $z(t) > 0$ and $U(t) \geq 0\ \forall t \geq 0$ as well as $\lim_{t\to+\infty} x(t) \approx x_0$, $\lim_{t\to+\infty} y(t) \approx y_0$ and $\lim_{t\to+\infty} z(t) = 0$ for both Case II and Case III, the positivity constraint of the system states and that of the dose concentration will be satisfied, the concentration of the HIV-1 viral load will be monotonic decreasing, as well as the asymptotical stability of the overall system will be guaranteed. This completes the proof. □

There are four parameters in the controlled system, namely, α, k_1, k_2 and k_3. Actually Theorem 1.1 is valid $\forall\, \alpha, k_1, k_2, k_3 > 0$. To design the value of α, since α is the threshold of the concentration of the HIV-1 viral load, in which the control strategy is switched from Case I to either Case II or Case III, and the rate of the change of the concentration of the HIV-1 viral load under the control strategy defined by Case I is faster than that defined by Case II and Case III, α should be small so that the concentration of the HIV-1 viral load will decrease at a faster rate. However, as $\frac{d}{dt}z(t) \neq 0$ when $z(t) = 0$, too small value of α would cause a rapid change of the state trajectory near the equilibrium point. Consequently, patients cannot stop the therapy. To obtain a balance between these two factors, we choose α to be a value equal to 59.5140% of $z(0)$ in our computer numerical simulation, that

is, 0.0179. For the values of k_1, k_2 and k_3, from (1.16), we see that $U(t)$ is small if k_1 and k_2 are small and k_3 is large. In the practical situation, $U(t)$ should be small so that the side effects of the medications are negligible. However, we can see from (1.18) that the rate of change of the concentration of the HIV-1 viral load is large if k_1 and k_2 are large and k_3 is small. As a fast response is preferred, there is a tradeoff between the strength of the side effects and the convergent rate of the system. To obtain a balance between these two factors, we propose to solve the problem via an optimization approach as follows:
For the control strategy defined by Case I, since

$$\begin{aligned}&\frac{d}{dt}U(t)\\&=\frac{dy(t)}{dt}\left[\frac{2k_2d}{k_3}\left(y(t)-\frac{y_0}{2}\right)-\frac{e^2(x(t))^2z(t)}{4f(y(t))^2}\right]+\frac{e^2}{4f}\left(\frac{2x(t)z(t)\frac{d}{dt}x(t)+(x(t))^2\frac{d}{dt}z(t)}{y(t)}\right)\end{aligned} \tag{1.24}$$

and $\frac{d}{dt}z(t)<0\ \ \forall t\geq 0\ \ \frac{d}{dt}x(t)<0$, if $\frac{d}{dt}y(t)>0$, $\frac{d}{dt}y(t)>0$ and $\frac{2k_2d}{k_3}\left(y(t)-\frac{y_0}{2}\right)<\frac{e^2(x(t))^2z(t)}{4f(y(t))^2}\ \ \forall z(t)>\alpha$, then we have $\frac{d}{dt}U(t)<0\ \ \forall z(t)>\alpha$. This implies that the maximum value of $U(t)$ will occur at $t=0$ if the above conditions are satisfied. Define an acceptable bound on the dose concentration as $U_{\max}$. Then k_1, k_2 and k_3 are designed such that $U(t)<U_{\max}\ \forall z(t)>\alpha$. That is:

$$\frac{e^2(x(0))^2z(0)}{4fy(0)}+\frac{k_2d}{k_3}\left(y(0)-\frac{y_0}{2}\right)^2+\frac{k_1bx_0^2}{4k_3}<U_{\max}. \tag{1.25}$$

Similarly, we define an acceptable bound on the rate of change of the concentration of the HIV-1 viral load and on the rate of change of $V(t)$ at $t=0$ as $\dot{z}_{\max}$ and $\dot{V}_{\max}$, respectively. From (1.18) to (1.19), we have

$$-\frac{z(0)}{4fy(0)}(ex(0)-2fy(0))^2-\frac{k_2d}{k_3}\left(y(0)-\frac{y_0}{2}\right)^2-\frac{k_1bx_0^2}{4k_3}<\dot{z}_{\max} \tag{1.26}$$

and

$$\begin{aligned}&-\frac{k_3(z(0))^2}{2fy(0)}(ex(0)-2fy(0))^2-2k_2c(y_0-y(0))^2-2k_1a(x_0-x(0))^2\\&-2k_1bz(0)\left(x(0)-\frac{x_0}{2}\right)^2-\frac{k_2dz(0)y_0^2}{2}<\dot{V}_{\max}\end{aligned}. \tag{1.27}$$

(1.25)–(1.27), as well as k_1, k_2 and k_3 being positive, forms a linear matrix inequality constraint on k_1, k_2 and k_3. For most of patients, $U_{\max}$ should be set around 0.37. Hence, we choose $U_{\max}$ accordingly. In order to have a fast response, $\dot{z}_{\max}$ and $\dot{V}_{\max}$ should be as small as possible, hence we choose $\dot{z}_{\max}$ and $\dot{V}_{\max}$ as -0.15 and -0.0095, respectively.

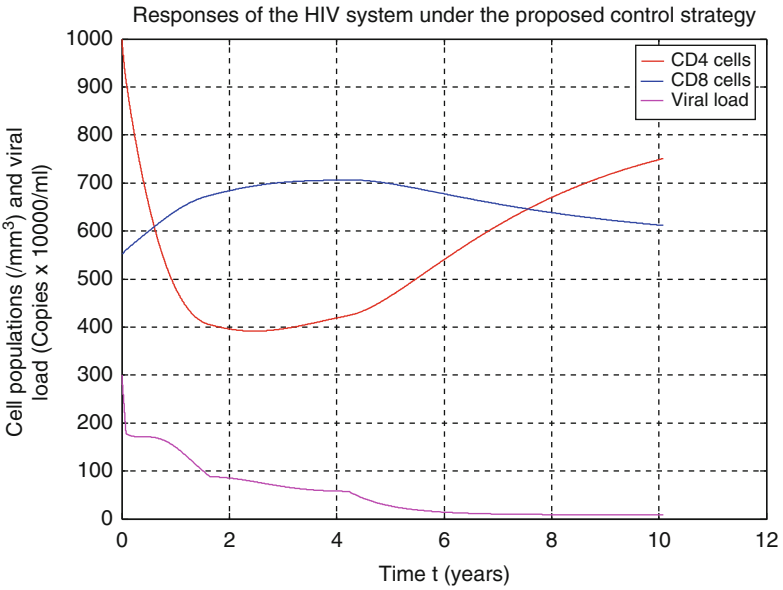

Fig. 1.10 State responses of the overall controlled system

Due to the tradeoff between the strength of the side effects and the convergent rate of the system, we formulate the following optimization problem as follows:

Problem (P)

$$\min_{(k_1,k_2,k_3)} \; k_1 + k_2 + k_3, \tag{1.28a}$$

subject to (1.25)–(1.27), (1.28b)

and k_1, k_2 and k_3 are positive. (1.28c)

Problem (P) is a standard linear programming problem, many existing solvers, such as Matlab optimization toolbox, can be employed for solving the problem. Moreover, as this optimization problem is convex, the obtained solution will be globally optimal if the feasible set is nonempty.

Figure 1.10 plots the controlled state responses with the system parameters and the initial condition being the same as those applied in Fig. 1.1, where α, k_1, k_2 and k_3 are determined based on the above discussion. It can be seen from Fig. 1.10 that the concentration of the HIV-1 viral load is monotonic decreasing, the system states asymptotically converge to the equilibrium point, as well as the concentrations of the CD4 lymphocyte population, the CD8 lymphocyte population and the HIV-1 viral load are non-negative and bounded. Hence, the proposed control strategy satisfies the positivity constraint. Figure 1.11 shows the corresponding dose concentration. It can be seen from Fig. 1.11 that the maximum dose concentration within the first 4 weeks is 0.37, which satisfies the specification. The dose concentration is monotonic decreasing within the first 4 weeks, but it drops dramatically to

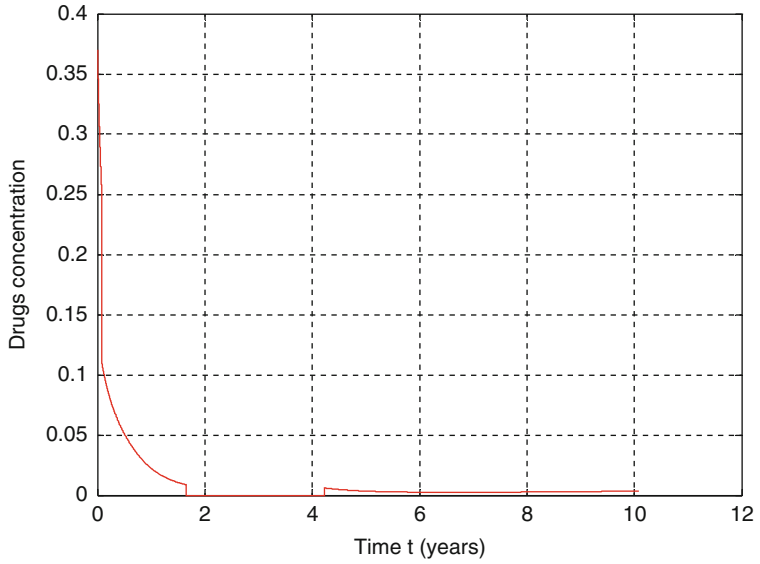

Fig. 1.11 Dose concentration

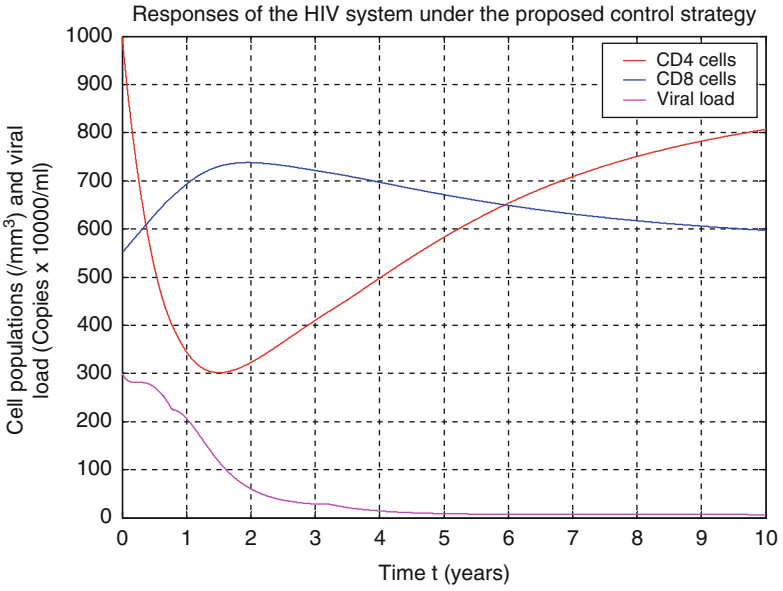

Fig. 1.12 State responses of the overall controlled system

low values after the 4th week because the system switches from the control strategy defined by Case I to that defined by Case II. Between the 85.7th and 219.7th weeks, the dose concentration is exactly equal to zero. This implies that the system

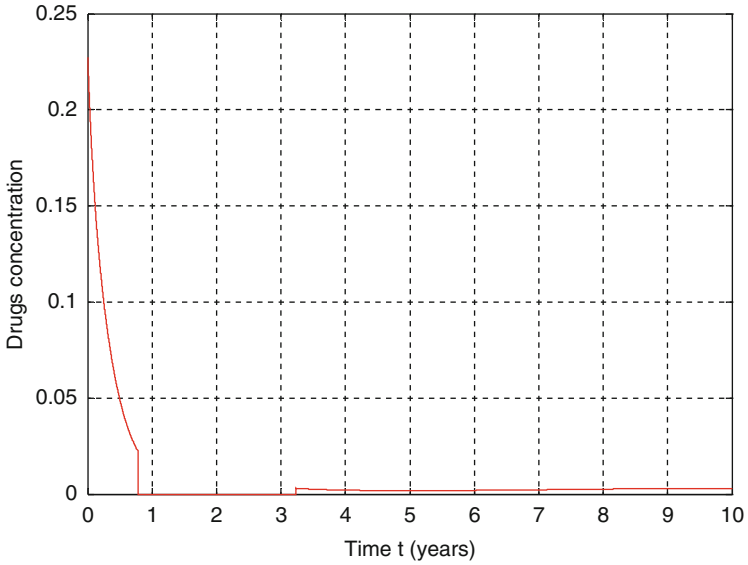

Fig. 1.13 Dose concentration

switches to the control strategy defined by Case III and no therapy is required. After the 219.7th week, the system switches back to the control strategy defined by Case II, so the dose concentration is nonzero and maintains at low level. To illustrate the effectiveness of the control strategy defined by Case I, Fig. 1.12 plots the controlled state responses of the system switched only between Case II and Case III. Figure 1.13 plots the corresponding dose concentration with the system parameters and the initial condition being the same as those applied in Fig. 1.1. It can be seen from the Fig. 1.13 that the concentration of the HIV-1 viral load decreases slowly while that based on the control strategy defined by Case I drops to the value equal to α quickly (at the 4th week).

1.5 Conclusions

In this chapter, we suggest to initiate the therapy based on both the concentration of the HIV-1 viral load and the ratio of the CD4 lymphocyte population to the CD8 lymphocyte population. If the concentration of the HIV-1 viral load is higher than a threshold, then the first type of therapy will be applied. If the concentration of the HIV-1 viral load is lower than or equal to the threshold and the ratio of the CD4 lymphocyte population to the CD8 lymphocyte population is greater than another threshold, then the second type of therapy will be applied. Otherwise, no therapy will be applied. The advantages of the proposed control strategy are that the therapy can be stopped under certain conditions, while the state variables of the overall

system is asymptotically stable with fast convergent rate, the concentration of the controlled HIV-1 viral load is monotonic decreasing, as well as the positivity constraint of the system states and that of the dose concentration is guaranteed to be satisfied.

References

1. Craig, I.K., Xia, X., Venter, J.W.: Introducing HIV/AIDS education into the electrical engineering curriculum at the university of Pretoria. IEEE Trans. Edu. **47**(1), 65–73 (2004)
2. Filter, R.A., Xia, X., Gray, C.M.: Dynamic HIV/AIDS parameter estimation with application to a vaccine readiness study in Southern Africa. IEEE Trans. Biomed. Eng. **52**(5), 784–791 (2005)
3. Xia, X., Moog, C.H.: Identifiability of nonlinear systems with application to HIV/AIDS models. IEEE Trans. Automat. Contr. **48**(2), 330–336 (2003)
4. Hammond, B.J.: Quantitative study of the control of HIV-1 gene expression. J. Theor. Biol. **163**, 199–221 (1993)
5. Wein, L.M., Zenios, S.A., Nowak, M.A.: Dynamic multidrug therapies for HIV: a control theoretic approach. J. Theor. Biol. **185**, 15–29 (1997)
6. De Boer, R.J., Perelson, A.S.: Target cell limited and immune control models of HIV infection: a compression. J. Theor. Biol. **190**, 201–214 (1998)
7. Wein, L.M., D'Amato, R.M., Perelson, A.S.: Mathematical analysis of antiretroviral therapy aimed at HIV-1 eradication or maintenance of low viral loads. J. Theor. Biol. **192**, 81–98 (1998)
8. Dixit, N.M., Perelson, A.S.: Complex patterns of viral load decay under antiretroviral therapy: influence of pharmacokinetics and intracellular delay. J. Theor. Biol. **226**, 95–109 (2004)
9. Strfford, M.A., Corey, L., Cao, Y., Daar, E.S., Ho, D.D., Perelson, A.S.: Modeling plasma virus concentration during primary HIV infection. J. Theor. Biol. **203**, 285–301 (2000)
10. Fishman, M.A., Perelson, A.S. Th1/Th2 cross regulation. J. Theor. Biol. **170**, 25–56 (1994)
11. Essunger, P., Perelson, A.S.: Modeling HIV infection of $CD4^+$ T-cell subpopulations. J. Theor. Biol. **170**, 367–391 (1994)
12. Berry, R.M., Nowak, M.A. Defective escape mutants of HIV. J. Theor. Biol. **171**, 387–395 (1994)
13. Lipsitch, M., Nowak, M.A.: The evolution of virulence in sexually transmitted HIV/AIDS. J. Theor. Biol. **174**, 427–440 (1995)
14. Nowak, M.A., May, R.M., Sigmund, K.: Immune responses against multiple epitopes. J. Theor. Biol. **175**, 352–353 (1995)
15. Courchamp, F., Pontier, D., Langlais, M., Artois, M.: Population dynamics of feline immunodeficiency virus within cat populations. J. Theor. Biol. **175**, 553–560 (1995)
16. Iwasa, Y., Michor, F., Nowak, M.A. Virus evolution within patients increases pathogenicity. J Theor. Biol. **232**, 17–26 (2005)
17. Nowak, M.A., Bonhoeffer, S., Shaw, G.M., May, R.M.: Anti-viral drug treatment: dynamics of resistance in free virus and infected cell populations. J. Theor. Biol. **184**, 203–217 (1997)
18. Wodarz, D., Lloyd, A.L., Jansen, V.A.A., Nowak, M.A.: Dynamics of macrophage and T cell infection by HIV. J. Theor. Biol. **196**, 101–113 (1999)
19. Gilchrist, M.A., Coombs, D., Perelson, A.S.: Optimizing within-host viral fitness: infected cell lifespan and virion production rate. J. Theor. Biol. **229**, 281–288 (2004)
20. Jafelice, R.M., Barros, L.C., Gomide, R.C.B.F.: Fuzzy set-based model to compute the life expectancy of HIV infected populations. IEEE Ann. Meet. Fuzzy Inform. Process. NAFIPS **1**, 314–318 (2004)

21. Ge, S.S., Tian, Z., Lee, T.H.: Nonlinear control of a dynamic model of HIV-1. IEEE Trans. Biomed. Eng. **52**(3), 353–361 (2005)
22. Campello de Souza, F.M.: Modeling the dynamics of HIV-1 and CD4 and CD8 lymphocytes. IEEE Eng. Med. Biol. **18**(1), 21–24 (1999)
23. Jeffrey, A.M., Xia, X., Craig, I.K.: When to initiate HIV therapy: a control theoretic approach. IEEE Trans. Biomed. Eng. **50**(11), 1213–1220 (2003)
24. Brandt, M.E., Chen, G.: Feedback control of a biodynamical model of HIV-1. IEEE Trans. Biomed. Eng. **48**(7), 754–759 (2001)
25. Craig, I.K., Xia, X.: Can HIV/AIDS be controlled? Applying control engineering concepts outside traditional fields. IEEE Contr. Syst. Mag. **25**(1), 80–83 (2005)
26. Ko, J.H., Kim, W.H., Chung, C.C.: Optimized structured treatment interruption for HIV therapy and its performance analysis on controllability. IEEE Trans. Biomed. Eng. **53**(3), 380–386 (2006)
27. Arora, D., Skliar, M., Roemer, R.B.: Minimum-time thermal dose control of thermal therapies. IEEE Trans. Biomed. Eng. **52**(2), 191–200 (2005)

Chapter 2
Fuzzy Knowledge Based GIS for Zonation of Landslide Susceptibility

J.K. Ghosh, Devanjan Bhattacharya, and Swej Kumar Sharma

2.1 Introduction

Landslides are among the major natural disasters in the world. It is a common phenomenon especially in a tectonically fragile and sensitive mountainous terrain like the lower Indian Himalayas [1]. Landslides cause loss of life and property, damage to natural resources, and hamper developmental projects like roads, dams, communication lines, bridges etc. The high susceptibility to landslides of the mountainous regions is mainly due to complex geological setting with the contemporary crustal adjustments, varying slopes and relief, heavy snow and rainfall along with ever-increasing human interference [2]. To take a quick and safer mitigation measure and strategic planning, identification of landslide prone areas and Landslide Susceptibility Zonation (LSZ) is important. A comparison of the distribution of different causative factors allows identifying the areas with varying landslide potential which is a complex task as the occurrence of landslide is dependent on many factors [3].

Several field-based hazard zonation techniques have been applied in the last few decades to the problem of landslide categorization [4]. These approaches have some drawbacks such as: repetitive coverage of area is difficult, manual overlay of thematic maps is a time consuming and tedious job, the extent of area covered is generally small, and in most of the cases landslide identification is close to road and thus interior parts are not covered. These problems are removed with the advent of satellite data and processing using geographical information system (GIS). It is now possible to efficiently and correctly collect and manipulate a variety of spatial

J.K. Ghosh • D. Bhattacharya (✉)
Indian Institute of Technology Roorkee, Civil Engineering Department, Roorkee, Uttarakhand 247667, India
e-mail: gjkumfce@iitr.ernet.in; devanjan@gmail.com

S.K. Sharma
Technologist, TATA Steel, Jamshedpur, India
e-mail: swejsharma@gmail.com

S. Banerjee et al. (eds.), *Applications of Chaos and Nonlinear Dynamics in Science and Engineering - Vol. 2*, Understanding Complex Systems,
DOI 10.1007/978-3-642-29329-0_2,

Table 2.1 Causative factors and their associated uncertainty

Factors	Uncertainty
Rock composition	Low
Rock structure	Intermediate
Ground water conditions	High
Slope geometry	Low
Geomorphic processes	Intermediate/High
Land use	Low

Table 2.2 A sample from KB showing the variables and assignments

Sl no.	Fuzzy variable	Subvariable	Weightage	Function based rating
1.	Slope Angle	$> 45°$	0.7	(Signum function)
		36°–45°		9
		26°–35°		7
		16°–25°		5
		$< 16°$		3
				1
2.	Landuse/landcover	Barren Land	0.4	(Gaussian function)
		Sparse Vegetation		9
		Moderate Vegetation		8
		Agricultural land		6
		Dense Forest		4
		Water		3
				1
3.	Relative relief	> 110 m	0.3	(Signum function)
		81–110 m		9
		51–80 m		7
		21–50 m		5
		< 21 m		3
				1

attributes such as geology, structure, surface cover, vegetation cover, the variability of terrain parameters and, distribution of landslides of an area. Moreover repetitive coverage, synoptic overview, availability of data in digital format, multi-spectral information and 3-D view capability and analysis is possible through these techniques [5–7].

Various geo-structural as well as causative-factor based approaches have been proposed for landslide susceptibility zonation [4,8]. In recent times, GIS modeling of landslide phenomena has taken precedence, and digital landslide zonation maps at various scales are now being generated. Expert systems for landslide have also been proposed [9,10]. But wherever consideration of landslide causative factors comes up, a comprehensive methodology of defining the degree of association of the contributory factors towards landslide is still to be offered.

For a case study and testing of the system, a detailed landslide susceptibility zonation map has been prepared of a slide prone area, using the proposed modern fuzzy based GIS techniques. The system works on a selection of appropriate fuzzy variables and membership values to integrate various data layers in GIS towards preparation of landslide susceptibility zonation map. The system is designed with a knowledge-base (KB) capable of managing uncertainties (Table 2.1) with the

knowledge (Table 2.2) stored in it. The difference with other such systems is in the use of fuzzy inferencing scheme, which effectively considers membership values of factors and sub-factors of landslide and the associated uncertainties.

2.2 Background

The requirement of this study is to evaluate multiple causative factors and their non-linear interactions to arrive at a decision. This could be achieved using a comprehensive input data set pertaining to the causative factors of the phenomena, fuzzy handling of the factors taking care of uncertainties and continuity, and multi-criteria evaluation of the contribution of the factors, all operating in a spatial domain of a GIS since the concerned problem is environmental, a geo-hazard called landslide. One common approach linking GIS, fuzzy logic and multi-criteria evaluation (MCE) [11,12] has been to utilize expert opinions on multiple criteria [13] and the resulting landslide maps are categorized into zones of "very low", "low", "medium", "high", and "very high" categories of susceptibility. In order to combine the selected landslide parameters and generate realistic output susceptibility maps, a GIS-based approach was developed using MCE and fuzzy sets. Each landslide contributory parameter affects the landslide process and final output maps so it is necessary to determine the relative importance of each parameter. The variabilities of the parameters have been captured effectively using fuzzy set membership functions [14] and the expert rules put down in the Indian Standard Code of Landslides [13] calculate the weight values for each parameter [15]. Final landslide susceptibility maps are then produced by using the weighted linear combination (WLC) method. A combination of the weighted parameterized maps using criteria weighting tools and criteria integration tools such as WLC lead to generate the landslide susceptibility maps [16].

2.2.1 GIS

Geographic information system (GIS) has rapidly developed from being a geographic database tool to now being able to provide sophisticated logical and mathematical analysis between multiple map layers. GIS gives the ability to store, handle, and transform spatial data efficiently and the integration of additional analytical techniques that can cope with multiple criteria problems has greatly enhanced the functionality of GIS. As a consequence, GIS is quickly evolving into a decision support system where decisions are made based on mapping outputs generated from the GIS analysis [16].

Spatial environmental problems with multiple criteria are easier to handle in a raster GIS. In the raster GIS data model, geographic space is represented as a grid of regular cells and each cell is coded with a single attribute value. With this raster GIS

model, remote sensing data based on the pixel can now be easily integrated into the GIS as the spatial conceptualization of the pixel is analogous to that of the grid cell. Once in the GIS, remote sensing data can then be subjected to advanced spatial data analysis techniques at various scales of analysis [16].

2.2.2 Fuzzy MCE

Fuzzy set theory was developed to deal with the inherent uncertainty of data by allowing membership of data belonging to a set along a continuous scale, rather than a crisp binary set membership [14]. Compared to linear scaling, fuzzy sets are a more realistic standardization approach because using a fuzzy set membership represents a specific relation between the criteria and possible outcomes [17,18]. When Boolean overlay is used, membership values are reduced to 0 and 1, which assumes crisp boundaries meaning the final outcome is identical to those of Boolean overlay [19]. Once the fuzzification has been defined, it is important to assign the relative importance of each evaluation criterion with respect to the overall objective of the problem [13]. By varying and prioritizing criteria, it is possible to generate compromising alternatives and thus rank alternative outcomes by the different expert opinions [13,20]. Since varying the relative importance of criteria reflects the knowledge of experts, it allows for sensitivity analysis and validation of weights and rankings of alternatives [16].

In multi-criteria evaluation (MCE) or multi-criteria analysis (MCA) the aim is to scrutinize a number of alternative possibilities given numerous criteria and conflicting objectives. In other words, MCE is primarily concerned with how to combine information from several criteria by imparting consistency and continuity to form a single index of evaluation. Criteria are the evidence that the outcome is based on and can be in the form of either factors or constraints [16]. A factor increases or decreases the suitability of a specific alternative for the activity under consideration, while a constraint provides exclusions and limits the alternatives under consideration. The degree of suitability of factors can range from either Boolean unsuitable to suitable (i.e. zeros and ones) to varying levels of certainty (i.e. a continuous scale from zero and one), which can be attributable to the type of data used and the subjectiveness of each expert [16].

MCE provides many advantages for use in defining landslide susceptibility. Landslide studies often use MCE techniques because the types of data available are commonly qualitative (from expert opinion) and quantitative (from observed relationships between parameters and landslides), therefore requiring a semi-quantitative method that incorporates both types of data [21]. These methods involve assigning weights to the different parameters affecting landslide susceptibility and many techniques exist for determining weights [22]. The most common approach involves obtaining expert opinion to assigning weights in a procedure known as Analytical Hierarchy Process (AHP) and then combining weights additively by weighted linear combination (WLC) to produce landslide susceptibility maps [23,24].

2.2.3 Model

The purpose of the multi-criteria evaluation approach used is to classify landslide susceptibility based on numerous conditioning factors in order to classify degree of landslide hazard in a region. The most common MCE factor aggregation procedure is known as weighted linear combination (WLC), where continuous parameters are standardized to a common numeric range, weights are calculated for each parameter, and then combined by weighted averaging for the final landslide susceptibility map.

Traditional GIS applications have used crisp Boolean sets, where spatial information represented discrete objects in a discrete definition; however, transition from membership to non-membership is rarely a step function. Rather, there is a gradual change represented as fuzzy sets and it is characterized by a membership function $f(\mu(x))$ where $\mu(x)\ \varepsilon\ [0,1]$ is a degree of membership to a particular set. In this study, fuzzy sets were used to represent individual landslide conditioning parameters and their relationship to landslide susceptibility. This membership for each landslide parameter is standardized on a scale from 0 to 1, where 0 represents the least susceptible and 1 represents the most susceptible. The fuzzy membership function used in this study is the sigmoidal or s-shaped membership, which follows a cosine function. The sigmoidal membership function is most commonly used in GIS applications. The broad methodology of the system consists of development of a knowledge-based system having capability to categorize a given region into different intensity of landslide and to provide a hazard map as its output. The system has been developed on the basic assumptions that (1) the terrain lies in mountainous region of a particular seismicity, (2) landslide occurrence is based on the causative factors as envisaged by the experts, and (3) a landslide will occur in the future under similar geo-environmental conditions, necessary as per the expert knowledge laid down in IS 14496 Part 2–1998 [13] on which the knowledge base is built. These weights must sum to one and they represent the relative importance and contribution that each parameter has on landslide susceptibility. The methodology lies in the superimposition of maps of causative factors of slope instability as proclaimed by the experts in the IS 14496 Part 2–1998 [13], namely lithology (rock and soil type), structure, slope morphometry, relative relief, land use land cover (LULC) and ground-water condition/rainfall. Each of these categories is associated with an uncertainty rating on the basis of its significance in causing instability (Table 2.1). An expert rating scheme as available in [13] for the various sub-categories of individual causative factors has been included in this study. A sample of the rating scheme is as Table 2.2.

Once landslide-conditioning parameters were standardized by fuzzy sets, the procedure used to combine information from multiple criteria for continuous factors is WLC, which can be expressed by $S = \sum w_i.x_i$, where S is the landslide susceptibility, w_i is the weight of the parameter i, and x_i is the criterion score of parameter i. In other words, parameters are combined by applying a weight to each followed by a summation of the results to yield a susceptibility map.

2.3 Implementation

For developing landslide hazard zonation map, different spatial data sets have been collected from different sources which are grouped as follows.

1. Topographic map of Haridwar district, lower Indian Himalayas.
2. Satellite data in the form of ASTER VNIR (very near infra-red) image of the same area.

The data preparation involved the use of IDRISI Andes 15.00 and Arc GIS 9.0 for GIS analysis, RSI ENVI 4.7 and ERDAS Imagine 8.5 for thematic map generation.

2.3.1 Data Integration and Analysis

The topographical and geological maps obtained in analogue form were converted into digital form through digitization. Together with these vector maps, the landuse/landcover map prepared by digital image processing of Aster image is used for GIS analysis, which has been used in this study to convert analogue map to digital form, extraction of different thematic maps (e.g. slope map, relative relief map), analysis of these maps and final landslide hazard zonation map preparation using fuzzy technique. The various layers which were used for mapping in the area include:

(a) Landuse/Landcover (LULC) map (Fig. 2.1).
(b) DEM (digital elevation model) derivatives—Slope Angle Map, Relative Relief Map (Figs. 2.2 and 2.3).

2.3.1.1 Generation of LULC Map

The landuse/landcover map of the study area has been prepared using following datasets:

(a) F.C.C. (false color composite) of ASTER VNIR image of Haridwar district.
(b) Google earth image of the same area.

Aster image (after pre-processing and registration) has been classified using supervised classification using ERDAS Imagine 8.5. The analysis has been performed using IDRISI Andes software. A total of six classes of landuse/landcover have been selected from the entire study area based on the above sources.

(a) Water. The water body appears cyanish-white in F.C.C.
(b) Dense Forest. Dense forest appears deep-red in F.C.C.

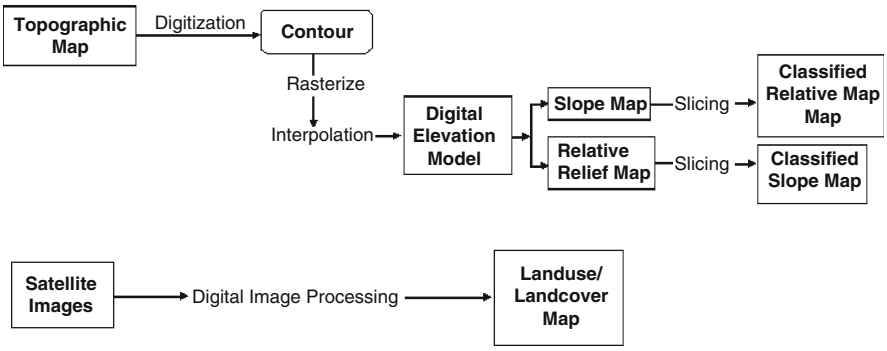

Fig. 2.1 Data preparation steps

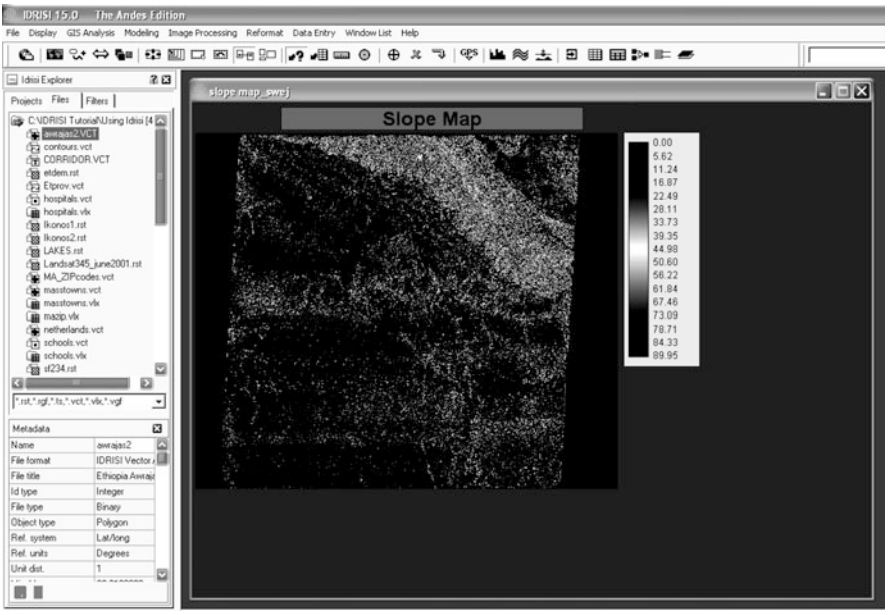

Fig. 2.2 DEM derivative slope map

(c) Moderate Vegetation. In ASTER F.C.C. moderate vegetation appears dull red.
(d) Agricultural Land. The agricultural fields are identified by pinkish-red in F.C.C. Mainly step-cultivation has been observed in this area surrounding small villages.
(e) Sparse Vegetation. The landcover where the vegetation density is very low with respect to the ground surface has been selected as sparse vegetation cover. Sparse vegetation shows up as dull red to green in F.C.C.
(f) Barren Land. Bluish-grey areas in F.C.C. have been chosen as barren areas.

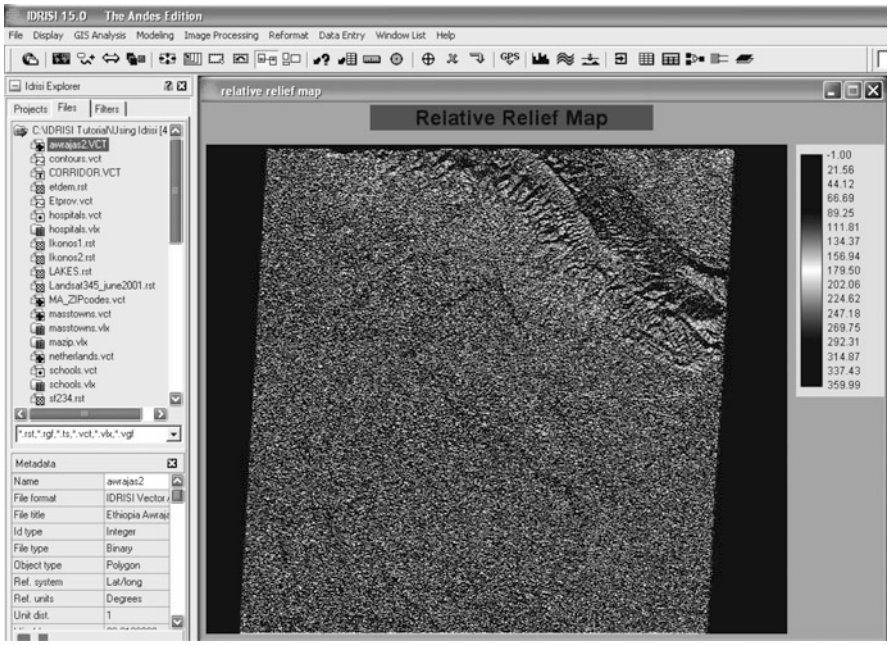

Fig. 2.3 DEM derivative relative relief map

Supervised Maximum Likelihood Classifier has been used to classify the Aster image. This classifier works on the principle that a given pixel may be assigned to that class for which it has the highest probability to be incorporated. It is assumed beforehand that the training data for each class in each band are normally (Gaussian) distributed.

The training data have been selected on per-pixel basis. The training pixels have been selected in such a way that these are well distributed over the entire area. From this training data set, various statistical parameters like mean and standard deviation to be used by Maximum Likelihood Classifier has been determined.

From the training data set statistics, each pixel in the image is allocated to one of the classes using the classifier. The pixel is allocated in that class with which it has the greatest probability of association. A classified map has thus been generated and area occupied by different classes has been calculated. An error matrix has been used to evaluate classification accuracy (Table 2.3). An unbiased set of test samples has been chosen from the reference data for assessment of classification accuracy. The overall accuracy has been calculated dividing the sum of diagonal element. The overall accuracy achieved is 69.13%.

Total Testing Pixels = 2112

Total Diagonal Elements = 1460

Overall Accuracy = (1460/2112) × 100 = 69.13%.

Table 2.3 Error matrix for assessment of classification accuracy

Classified data	Reference data (training set data)							
		Water	Dense forest	Moderate vegetation	Agricultural land	Sparse vegetation	Barren land	Total pixels
	Water	387	3	0	0	3	9	402
	Dense forest	0	288	51	3	27	0	369
	Moderate vegetation	0	39	213	9	69	33	363
	Agricultural land	0	0	9	228	9	6	252
	Sparse vegetation	0	57	117	21	207	0	402
	Barren land	9	3	21	12	42	237	324
	Total pixels	396	390	411	273	357	285	2,112

2.3.1.2 Generation of DEM Derivatives

We have generated a contour map by digitizing the topographic map of Haridwar and its surrounding area using Arc GIS 9.0. Then the DEM has been generated using the contours and imported to IDRISI Andes for LSZ map preparation. Using the *Surface Analysis* in IDRISI Slope and Relative Relief maps have been created. Slope is an important parameter for stability consideration. The slope at any point is a gradient between the centre and neighborhood cell with maximum or minimum elevation. A slope map thus is a raster map in which the attribute of each pixel denotes the slope at a particular location. The slope map thus derived from DEM shows a range of variation of slope from 0° to 87° in the study area. It has been classified using "slicing" function into five classes as follows.

Slope Classes— $< 16°$ $> 16°$—$25°$ $26°$—$35°$ $36°$—$45°$ $> 45°$

Relative relief is defined as the difference in maximum and minimum elevation values within an area or facet. In GIS the relative relief has been calculated by generating two maps using 3×3 rank order filter—one with maximum elevation and other with minimum elevation within a 3×3 window. The minimum elevation map is subtracted from maximum elevation map through "calculator" function to get to a relative relief map. The range of relative relief is found to be 0–174 m. On the basis of histogram and analyzing the spatial distribution, the map has been classified into five classes.

Relative Relief Classes— $>$ 110 m; 81–110 m; 51–80 m; 21–50 m; $<$ 21 m.

2.4 Fuzzy System

The architecture of the proposed system has three functional modules—a GIS based Input Module, a Fuzzy Expert Module and an Output Module, wherein the expert module combines the knowledge module and the fuzzy inference module as sub-modules.

2.4.1 Proposed Architecture of the System

2.4.1.1 Input Module

The input module has been developed in IDRISI to accept layers of information of causative factors of landslides.

Table 2.4 Landslide susceptibility zonation on the basis of output fuzzy membership functions

Zone	Fuzzy membership function	Description	Function
I	<0.1	Very low hazard	Sigmoidal function
II	0.1–0.4	Low hazard	Sigmoidal function
III	0.4–0.6	Moderate hazard	Sigmoidal function
IV	0.6–0.75	High hazard	Sigmoidal function
V	>0.75	Very high hazard	Sigmoidal function

2.4.1.2 Expert Module

Knowledge-Base

The KB is derived from ground data and the Indian Standard Code [13]. The six causative factors with their sub-factors and ratings are stored in the Knowledge module, a sub-module of the expert module. Membership values and ratings are also defined in this sub-module with the appropriate fuzzy functions [14] as shown in Tables 2.2 and 2.4. The weighting and rating system is based on relative importance of various causative factors and the actual field knowledge on them. Table 2.2 describes the membership value and ratings given to each layer and their classes respectively. Theses membership and rating values have been re-adjusted using trial and error method [11,12] by matching it with the ground truth [13].

(a) Slope Angle: A very important parameter in landslide activity, in this area, the slopes greater than 45° are highly unstable [13,25].
(b) Landuse/landcover: Among all the classes, barren land is most prone to landslide activity. Hence, the highest rating is given to this class. As the vegetation density increases, the stability of slope increases [26].
(c) Relative Relief: The higher relative relief leads to greater landslide susceptibility. Therefore, the rating is given in increasing order as the relative relief increases.

Inference Engine

Computation of Landslide Hazard Index (LHI) is performed in this sub-module where LHI is given by:

$$\text{LHI} = \sum \text{weightage} \times \text{data layer (attribute)} = \sum \mu * X$$

For this study, the Landslide Hazard Index = [0.7 * Slope Angle + 0.4 * Landuse/landcover + 0.3 * Relative Relief]

2.4.1.3 The Output Module

The Output Module provides a grey coded (consisting of five different grey scale levels for the five levels of susceptibilities) digital map of the intensity of landslide

susceptibility prevalent in the region of study, and a two-level binary map showing the entire region in two categories, susceptible or not susceptible.

2.4.2 Methodology

A decision support approach called Multi-Criteria Evaluation (MCE) using fuzzy system has been used [27]. A decision is a choice between alternatives. The basis for a decision is known as a criterion. In a Multi-Criteria Evaluation, an attempt is made to combine a set of criteria to achieve a single composite basis for a decision according to a specific objective. For example, a decision may need to be made about what areas are the most suitable for industrial development. Criteria might include proximity to roads, slope gradient, exclusion of reserved lands, and so on [28]. Through a Multi-Criteria Evaluation, these criteria images representing suitability may be combined to form a single suitability map from which the final choice will be made [29].

Criteria may be of two types: factors and constraints. Factors are generally continuous in nature such as slope gradient or road proximity factors; they indicate the relative suitability of certain areas [27,30]. Constraints, on the other hand, are always boolean in character such as reserved lands constraint. They serve to exclude certain areas from consideration. Factors and constraints can be combined in the MCE module and are characterized by different levels of control over tradeoff between factors and the level of risk assumed in the combination procedure where tradeoff is the degree to which one factor can compensate for another; how they compensate is governed by a set of factor weights sometimes called tradeoff weights [31,32]. Factor weights are given for each factor such that all factor weights, for a set of factors, sum to one; they indicate the relative importance of each factor to the objective under consideration. A factor with a high factor/tradeoff weight may compensate for low suitability in other factors that have lower factor/tradeoff weights [19,27].

In addition to tradeoff, any MCE is also characterized by some level of assumed risk that will strongly influence the final suitability map. A low risk analysis is one where the area considered most suitable in the final result is minimized since it must be highly suitable in all factors. A high risk analysis is one where the area considered most suitable in the final result will be maximized since any area that is highly suitable for any one factor will be considered highly suitable in the result.

Constraints must be in byte or binary format and should be boolean maps with zeros in areas that are excluded from consideration and ones elsewhere. Factors must also be in byte or binary format with a standard scaling (i.e., all factors must use the same scaling system). For example, they might all have values that range from 0 to 255, or 0 to 99 which is to standardize the continuous factors into the desired range using one of a number of set membership functions (linear, sigmodal, J shaped, or user defined); it also allows for increasing, decreasing, or symmetric functions. Once the standardized factor images have been created, a set of weights

have been developed that indicate the relative importance of each factor to the decision under consideration. These weights must sum to one.

By adding all the weight maps the Landslide Hazard Index [13,33,34] has been found to vary between the range of 0–1. Gamma values assigned for each gamma combination operation play the most important role on the output fuzzy membership functions [35]. Owing to the constant value for gamma, the outputs are essentially based on the fuzzy membership functions assessed to factor maps classes. Gamma values >0.94 moves the output fuzzy membership function values to high and very high landslide susceptibilities and eliminates the moderate susceptible zones. It is suggested to examine different values of gamma and evaluate the results based on the relationship between known landslides and the susceptibility zones. Landslide susceptibility zonation on the basis of output fuzzy membership functions is shown in Table 2.4.

2.5 Results

The present study is an attempt to prepare a detailed landslide susceptibility zonation map of Haridwar district, lower Himalayan foothills in India. From the distribution of various susceptibility classes in the landslide susceptibility zonation map (Figs. 2.4 and 2.5), following conclusions can be drawn.

(a) Very high susceptibility zone has a very high possibility of slope failure of as much as 9–12% of the area of study.
(b) High susceptibility zone occupies approximately 20–25% of the total study area.
(c) The study area is mainly falling in low hazard and moderate hazard zone.

A comparison between the landslide susceptibility class zones on the map and the landslides was carried out. All the three major known landslides were located in the high and very high susceptible zones.

2.6 Discussion and Conclusion

The fuzzy logic non-linear modeling approach used in this paper provides a flexible method with which to include expert opinion in developing an inference network. A variety of fuzzy operators are used to examine different combinations and produce an intermediate map, or add any new data layer to the model and to test its affect on the final possibility map. Because the fuzzy membership functions assessed for factor maps were mainly extracted from the field data, the procedure followed during the study should find locations of the known landslides. In order to control the performance of the produced susceptibility map, a comparison between the landslide susceptibility class zones on the map and the landslides has been carried out. All the three major known landslides were located in the high and very

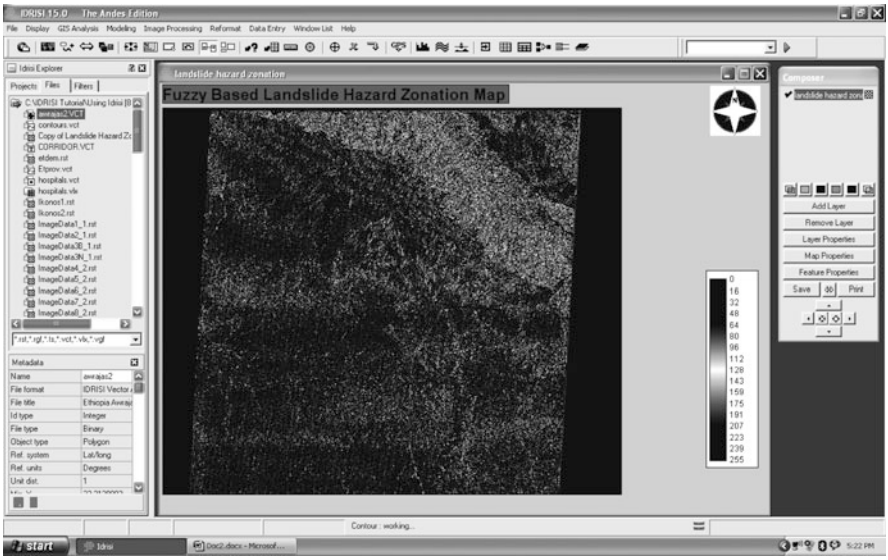

Fig. 2.4 Landslide susceptibility zonation map

high susceptible zones. This could be an acceptable result for a medium-scale landslide susceptibility map, and the use of fuzzy sigmoidal function in the generation of the landslide susceptibility map seems to be a reliable approach.

This landslide susceptibility map can be used as a planning tool but would rather not be recommended for individual site specific evaluation. Areas within the high and very high susceptibility categories should require further study by engineering geologists before development to determine the extent of possibly unstable conditions. Because the membership function approach has the nature of fuzzy rather than present-absent, or binary pattern, it may identify localities which previously have not been recognized by other methods, such as those based on known deposits. The fuzzy maps also have valuable data for other analysis methods, which apply some or all of the quantified factors. The fuzzy logic method is objective and repeatable, and can utilize varying reliability data, which commonly occur in geological descriptions.

Gamma values assigned for each gamma combination operation play the most important role on the output fuzzy membership functions. Owing to the constant value for gamma, the outputs are essentially based on the fuzzy membership functions assessed to factor maps classes. Gamma values >0.94 moves the output fuzzy membership function values to high and very high landslide susceptibilities and eliminates the moderate susceptible zones. It is suggested to examine different values of gamma and evaluate the results based on the relationship between known landslides and the susceptibility zones.

Apart from the spatial extents of landslide susceptibility that have come out with the result, it is seen that the very high and high hazard zones are mainly concentrated around the thrust.

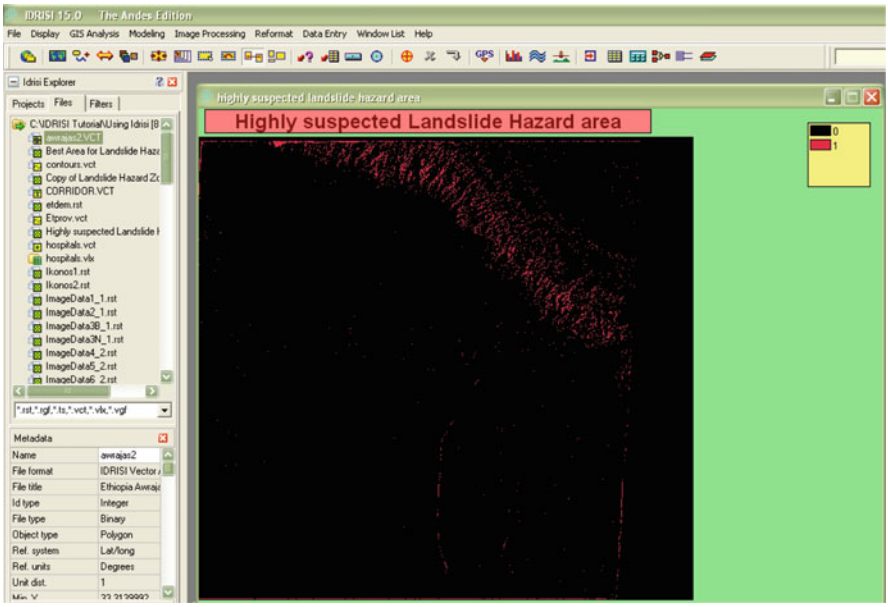

Fig. 2.5 Two level boolean landslide susceptibility zonation map showing high hazard

This method of landslide hazard zonation mapping depends mainly on the weights assigned to the parameters responsible for landslide. These weights and rating values should be re-adjusted using trial and error method. Here, from the above results it can be stated that the weighting-rating system adopted in this study is quite suitable for the study area. The landslide hazard zonation map depicts relative risk susceptibility of areas to landslides under natural conditions. The changes in natural environment (e.g. by human interference like road construction, deforestation etc.) may change the risk susceptibility of the area in terms of landslides.

Future scope of work is immense considering the fact that enhancements are possible in the knowledge base, use of still more efficient fuzzy operators and better implementation of non-linearity, and generalization of the system over national or even international extents. If susceptibility maps of larger scales are to be produced then site-specific parameters need to be taken into account which is another area of work.

References

1. Ghosh, J.K., Suri, S: A knowledge based system for assessment of landslide hazard. In Proceedings of the Indian Geotechnical Conference, vol. 1, pp. 393–396. IGS, Ahmedabad (2005)
2. Nagarajan, R., Mukherjee, A., Roy, A., Khire, M.: Temporal remote sensing data and GIS application in landslide hazard zonation of part of Western Ghat, India. Int. J. Rem. Sens. **19**(4), 573–585 (1998)

3. Park, N.W., Chi, K.H.: A probabilistic approach to predictive spatial data fusion for geological hazard assessment. Proc. Symp. IEEE IGARSS, France, **4**, 2425–2427 (2003)
4. Guzzetti, F., Reichenbach, P., Cardinali, M., Galli, M., Ardizzone, F.: Probabilistic landslide hazard assessment at the basin scale. Geomorphology **72**, 272–299 (2005)
5. Lee, S.I., Ryu, J., Min, K., Choi, W., Won, J.: Development and application of landslide susceptibility analysis techniques using GIS. In Proceedings of the Symposium on Geoscience and Remote Sensing, pp. 319–321. IEEE IGARSS, Honolulu (2000)
6. Muthu, K., Petrou, M.: Landslide-hazard mapping using an expert system and a GIS. IEEE Trans. Geosc. Rem. Sens. **45**(2) (2007)
7. van Westen, C.J.: Digital geomorphological landslide hazard mapping of Alpago, Italy. Int. J. Appl. Earth Obs. Geoinfo.(JAG), **2**(1), 51–59 (2000)
8. Piantanakulchai, N.: Analytic network process model for landslide hazard zonation. Eng. Geol. **85**(3–4), 281–294 (2006)
9. Parsons, S.: Current approaches to handling imperfect information in data and knowledge bases. IEEE Trans. Know. Data Eng. **8**, 353–372 (1996)
10. Ghosh, J.K., Bhattacharya, D.: A knowledge based landslide susceptibility zonation system. Am. Soc. Civ. Eng. J. Comp. Civ. Eng. **24**(4), 325–334 (2010), available online at: link.aip.org/link/doi/10.1061/(ASCE)CP.1943-5487.0000034
11. Jankowski, P.: Integrating geographical information systems and multiple criteria decision-making methods. Int. J. Geogr. Inform. Syst. **9**(3), 251–273 (1995)
12. Ng, K.C., Abramson, B.: Uncertainty management in expert systems. IEEE Expert Syst. **5**(2), 29–48 (1990)
13. I.S. 14496, [Part 2]: Indian Standard Preparation of Landslide Hazard Zonation Maps in Mountainous Terrains – Guidelines. Part 2 Macro-Zonation. Bureau of Indian Standard, New Delhi, pp. 1–19, 1998
14. Zadeh, L.A.: Fuzzy sets. Inform. Contr. **8**(3), 338–353 (1965)
15. Gupta, P., Anbalagan, R.: Landlide hazard zonation, mapping of Tehri-Pratapnagar area, Garhwal Himalayas. J. Rock Mech. Tunn. Tech., India. **1**(1), 41–58 (1995)
16. Lai, T.: Modelling spatial dynamics of landslides: integration of GIS-based cellular automata and multicriteria evaluation methods. M.Sc. Thesis, Department of Geography, Simon Fraser University (2011)
17. Wood, L.J., Dragicevic, S.: GIS-based multicriteria evaluation and fuzzy sets to identify priority sites for marine protection. Biodiversity Conservation **16**(9): 2539–2558 (2007)
18. Gorsevski, P.V., Jankowski, P.: An optimized solution of multi-criteria evaluation analysis of landslide susceptibility using fuzzy sets and Kalman filter. Comput. Geosci. **36**(8), 1005–1020 (2010)
19. Jiang, H., Eastman, J.R.: Application of fuzzy measures in multi-criteria evaluation in GIS. Int. J. Geogr. Inform. Sci. **14**(2), 173–184 (2000). www.tandf.co.uk/journals/tf/13658816.html
20. Malczewski, J.: GIS-based multicriteria decision analysis: a survey of the literature. Int. J. Geogr. Inform. Sci. **20**(7), 703–726 (2006)
21. Ayalew, L., Yamagishi, H.: The application of GIS-based logistic regression for landslide susceptibility mapping in the Kakuda-Yahiko Mountains, Central Japan. Geomorphology **65**(1–2), 15–31 (2005)
22. Akgun, A., Bulut, F.: GIS-based landslide susceptibility for Arsin- Yomra (Trabzon, North Turkey) region. Environ. Geol. **51**(8), 1377–1387 (2007)
23. Komac, M.: A landslide susceptibility model using the analytical hierarchy process method and multivariate statistics in penialpine Slovenia. Geomorphology **74**(1–4), 17–28 (2006)
24. Park, N.W., Chi, K.H., Kwon, B.D.: Application of fuzzy set theory for spatial prediction of landslide hazard. In Proceedings of the Symposium on Geoscience and Remote Sensing, vol. 5, pp. 2988–2990. IEEE IGARSS, Alaska (2004)
25. Ostir, K., Veljanovski, T., Podobnikar, T., Stancic, Z.: Application of satellite remote sensing in natural hazard management. Int. J. Rem. Sens. **24**(20), 3983–4002 (2003)

26. Yesilnacar E., Suzen, M.L.: A land-cover classification for landslide susceptibility mapping by using feature components. Int. J. Rem. Sens. **27**(2), 253–275 (2006)
27. Alemayehu, D.: Assessment of flood risk in Dire-Dawa town. Master's Thesis, Addis Ababa University, pp. 12–13, link: danielalemayehu.pdf (2007)
28. van Westen, C.J., van Asch, T.W.J., Soeters, R.: Landslide hazard and risk zonation-why is it still so difficult? Bull. Eng. Geol. Env. **65**, 167–184 (2006)
29. Bonham-Carter, G.F.: Geographic Information Systems for Geoscientists, Modeling with GIS, p. 398. Pergamon, Oxford (1994)
30. Lee, S.: Application and verification of fuzzy algebraic operators to landslide susceptibility mapping. Environ. Geol. **52**, 615–623 (2007)
31. Pistocchi, A., Luzi, L., Napolitano, P.: The use of predictive modeling techniques for optimal exploitation of spatial databases: a case study in landslide hazard mapping with expert system-like methods. Environ. Geol. **58**, 251–270 (2002)
32. Pradhan, B., Lee, S.: Landslide risk analysis using artificial neural network model focussing on different training sites. Int. J. Phys. Sci. **4**(1), 001–015 (2009)
33. Murgante, B., Casas, G.L. G.I.S. and fuzzy sets for the land suitability analysis. Lecture Notes in Computer Science LNCS, vol. 3044, pp. 1036–1045, ISSN: 0302–9743. Springer-Verlag, Berlin (2004)
34. Petry, F.E., Robinson, V.B., Cobb, M.A. (eds.): Fuzzy Modeling with Spatial Information for Geographic Problems, XII, 338 p. Springer, Berlin (2005)
35. Baja, S., Chapman, D.M., Dragovich, D.: A conceptual model for defining and assessing land management units using a fuzzy modeling approach in GIS environment. Environ. Manag. **29**(5), 647–661 (2002). Springer, New York

Chapter 3
An Output Feedback Control Design for H_∞ Synchronization of Master-Slave Markovian Jump Systems with Time-Varying Delays

Hamid Reza Karimi

3.1 Introduction

In recent years, more attention has been devoted to the study of stochastic hybrid systems, where the so-called Markov jump systems. These systems represent an important class of stochastic systems that is popular in modeling practical systems like manufacturing systems, power systems, aerospace systems and networked control systems that may experience random abrupt changes in their structures and parameters [1–4]. Random parameter changes may result from random component failures, repairs or shut down, or abrupt changes of the operating point. Many such events can be modeled using a continuous time finite-state Markov chain, which leads to the hybrid description of system dynamics known as a Markov jump parameter system [5–11]. Furthermore, the delay effects problem on the stability of systems is a problem of recurring interest since the delay presence may induce complex behaviors for the schemes, see for instance [12, 13]. The problem of filtering for state delayed systems with Markovian switching is proposed in [14–18]. The problem of robust mode-dependent delayed state feedback H_∞ control is investigated for a class of uncertain time-delay systems with Markovian switching parameters and mixed discrete, neutral and distributed delays in [19]. Moreover, the sliding mode control problem for uncertain systems with time delays and stochastic jump systems are also investigated in [20–28], respectively. Recently, the problem of sliding mode control for a class of nonlinear uncertain stochastic systems with Markovian switching is studied in [29]. More recently, in [30], sliding mode control of nonlinear singular stochastic systems with Markovian switching is proposed.

H.R. Karimi (✉)
Department of Engineering, Faculty of Engineering and Science, University of Agder, N-4898 Grimstad, Norway
e-mail: hamidrk@uia.no

S. Banerjee et al. (eds.), *Applications of Chaos and Nonlinear Dynamics in Science and Engineering - Vol. 2*, Understanding Complex Systems,
DOI 10.1007/978-3-642-29329-0_3,

On another research front line, synchronization is a basic motion in nature that has been studied for a long time, ever since the discovery of Christian Huygens in 1665 on the synchronization of two pendulum clocks. The results of chaos synchronization are utilized in biology, chemistry, secret communication and cryptography, nonlinear oscillation synchronization and some other nonlinear fields. The first idea of synchronizing two identical chaotic systems with different initial conditions was introduced by Pecora and Carroll in [31], and the method was realized in electronic circuits. The methods for synchronization of the chaotic systems have been widely studied in recent years, and many different methods have been applied theoretically and experimentally to synchronize chaotic systems; see for instance [32–35]. On the synchronization problems of systems with time-delays and nonlinear perturbation terms, we see that there have been some research works; see for instance [36–41] and the references therein. So the development of synchronization methods for master–slave systems with Markovian switching parameters and time-varying delays is important and has not been fully investigated in the past and remains to be important and challenging. This motivates the present study.

In this chapter, the problem of output feedback control design for exponential H_∞ synchronization is studied for a class of master–slave systems with both discrete and distributed time-delays and Markovian switching parameters. Using an appropriate Lyapunov–Krasovskii functional, some delay-dependent sufficient conditions and a synchronization law which include the master–slave parameters are established for designing a delay-dependent mode-dependent sliding mode exponential H_∞ synchronization control law in terms of linear matrix inequalities (LMIs). The output feedback controller guarantees the H_∞ synchronization of the two coupled master and slave systems regardless of their initial states. A numerical example is given to show the effectiveness of the method.

The rest of this chapter is organized as follows. Section 3.2 formulates the exponential H_∞ synchronization problem of the master and slave systems with Markovian switching parameters and mixed discrete and distributed time-varying delays. In Sect. 3.3, both H_∞ performance analysis and output feedback control design are presented for the system under consideration. In Sect. 3.4, computer simulations are provided to demonstrate the effectiveness of the proposed synchronization scheme. Finally, conclusions are presented in Sect. 3.5.

Notation: The notations used throughout the chapter are fairly standard. I and 0 represent identity matrix and zero matrix; the superscript $'T'$ stands for matrix transposition. $\|.\|$ refers to the Euclidean vector norm or the induced matrix 2-norm. $\mathrm{diag}\{...\}$ represents a block diagonal matrix and the operator $\mathrm{sym}(A)$ represents $A + A^T$. Let $\Re^+ = [0, \infty)$ and $\varepsilon\{.\}$ denotes the expectation operator with respect to some probability measure $\mathcal{P}$. If $x(t)$ is a continuous $\Re^n$-valued stochastic process on $t \in [-\kappa, \infty)$, we let $x_t = \{x(t+\theta) : -\kappa \le \theta \le 0\}$ for $t \ge 0$ which is regarded as a $C([-\kappa, 0); \Re^n)$-valued stochastic process. The notations $\mathcal{g}_{ji}$ stand for $\mathcal{g}_j(i)$. The notation $P > 0$ means that P is real symmetric and positive definite; the symbol $*$ denotes the elements below the main diagonal of a symmetric block matrix.

3.2 Problem Description

Consider a model of master and slave systems with Markovian switching parameters and mixed discrete and distributed time-varying delays in the form of

$$\begin{cases} \dot{x}_m(t) = A_1(r(t))x_m(t) + A_2(r(t))x_m(t-h(t)) + A_3(r(t)) \int\limits_{t-\tau(t)}^{t} x_m(s)ds, \\ x_m(t) = \phi(t), \qquad t \in [-\kappa, 0], \\ y_m(t) = C(r(t))x_m(t), \\ z_m(t) = C_1(r(t))x_m(t) + C_2(r(t))x_m(t-h(t)) + C_3(r(t)) \int\limits_{t-\tau(t)}^{t} x_m(s)ds \end{cases} \tag{3.1}$$

$$\begin{cases} \dot{x}_s(t) = A_1(r(t))x_s(t) + A_2(r(t))x_s(t-h(t)) + A_3(r(t)) \int\limits_{t-\tau(t)}^{t} x_s(s)ds \\ \qquad + B(r(t))u(t) + D(r(t))w(t), \\ x_s(t) = \varphi(t), \qquad t \in [-\kappa, 0], \\ y_s(t) = C(r(t))x_s(t), \\ z_s(t) = C_1(r(t))x_s(t) + C_2(r(t))x_s(t-h(t)) + C_3(r(t)) \int\limits_{t-\tau(t)}^{t} x_s(s)ds \end{cases} \tag{3.2}$$

where $\kappa := \max\{h_M, \tau_M\}$, $x_m(t), x_s(t)$ are the $n \times 1$ state vector of the master and slave systems, respectively, $w(t)$ is the disturbance and $u(t)$ is the $r \times 1$ control input. $z_m(t)$ and $z_s(t)$ are the controlled outputs of the drive and the response system, respectively.

$A_1(r(t)), A_2(r(t)), A_3(r(t)), B(r(t)), C_1(r(t)), C_2(r(t)), C_3(r(t)), D(r(t))$ are matrix functions of the random jumping process $\{r(t)\} \cdot \{r(t), t \geq 0\}$ is a right-continuous Markov process on the probability space which takes values in a finite space $S = \{1, 2, \ldots, s\}$ with generator $\Pi = [\pi_{ij}](i, j \in S)$ given by

$$P\{r(t+\Delta) = j | r(t) = i\} = \begin{cases} \pi_{ij}\Delta + o(\Delta), & \text{if } i \neq j \\ 1 + \pi_{ii}\Delta + o(\Delta), & \text{if } i = j \end{cases} \tag{3.3}$$

where $\Delta > 0$, $\lim_{\Delta \to 0} o(\Delta)/\Delta = 0$ and $\pi_{ij} \geq 0$, for $i \neq j$, is the transition rate from mode i at time t to mode j at time $t + \Delta$ and $\pi_{ii} = -\sum\limits_{j=1, j \neq i}^{j=s} \pi_{ij}$. The vector valued initial functions $\phi(t)$ and $\varphi(t)$ are continuously differentiable functionals. $h(t)$ and $\tau(t)$ denote the time-varying bounded delays satisfying

$$0 < h(t) \leq h_M, \quad \dot{h}(t) \leq h_D \tag{3.4a}$$

$$0 < \tau(t) \leq \tau_M, \quad \dot{\tau}(t) \leq \tau_D < 1 \tag{3.4b}$$

Definition 3.1. The Markovian systems (3.1)–(3.2) are said to be globally exponentially stable in the mean square sense if, when $u(t) = 0$, for any finite $\varphi(t)$, $\phi(t) \in \Re^n$ defined on $[-\kappa, 0]$, and $r_0 \in S$ the following condition is satisfied

$$\varepsilon\left\{\|e(t)\|^2\right\} \leq ce^{-\alpha t} \sup_{-\kappa \leq S \leq 0} \|e(s)\|^2, \quad t > 0$$

where $e(t) = x_m(t) - x_s(t)$ is the synchronization error of the master and slave systems (3.1)–(3.2) system from initial system state $\phi(0) - \varphi(0)$ and initial mode r_0, and c is a positive constant. Constant α is said to be the degree of exponential synchronization.

Definition 3.2. The H_∞ performance measure of the system (3.1)–(3.2) is defined as $J_\infty = \varepsilon\big(\int_0^\infty [Z_e(t)^T Z_e(t) - \gamma^2 W^T(t) W(t)]dt\big)$, where $Z_e(t) = Z_m(t) - Z_s(t)$ and the positive scalar γ is given.

Remark 3.1. The model (3.1)–(3.2) can describe a large amount of well-known dynamical systems with time-delays, such as the delayed Logistic model, the chaotic models with time-delays and the artificial neural network model with discrete time-delays. In real application, these coupled systems can be regarded as interacting dynamical elements in the entire system, such as physical particles, biological neurons, ecological populations, genic oscillations, and even automatic machines and robots. A feasible coupling design for successful synchronization leads us to fully command the intrinsic mechanism regulating the evolution of real systems, to fabricate emulate systems, and even to remotely control the machines and nodes in networks with large scales.

Assumption 3.2. *The full state variables* $x_s(t)$ and $x_m(t)$ *are available for measurement.*

Now, it is required to synchronize the slave system with the master system at the same time. Let $\hat{e}(t) = e^{\alpha t} e(t)$ and α is called the exponential decay rate. Then the error dynamics, namely synchronization error system, can be expressed by

$$\begin{cases} \dot{\hat{e}}(t) = (A_1(r(t)) + \alpha I)\hat{e}(t) + e^{\alpha h(t)} A_2(r(t))\hat{e}(t - h(t)) + A_3(r(t)) \int_{t-\tau(t)}^{t} e^{\alpha(t-s)}\hat{e}(s)ds \\ \qquad - B_1(r(t))\hat{u}(t) - B_2(r(t))\hat{w}(t), \\ \hat{e}(t) = e^{\alpha t}(\phi(t) - \varphi(t)), \qquad t \in [-\kappa, 0], \\ \hat{y}_e(t) = C(r(t))\hat{e}(t), \\ \hat{z}_e(t) = D_1(r(t))\hat{e}(t) + e^{\alpha h(t)} D_2(r(t))\hat{e}(t - h(t)) \\ \qquad + D_3(r(t)) \int_{t-\tau(t)}^{t} e^{\alpha(t-s)}\hat{e}(s)ds \end{cases} \tag{3.5}$$

where $\hat{u}(t) = e^{\alpha t}u(t)$, $\hat{w}(t) = e^{\alpha t}w(t)$, $\hat{y}(t) = e^{\alpha t}y(t)$, $\hat{z}_e(t) = \hat{z}_m(t) - \hat{z}_s(t) = e^{\alpha t}(z_m(t) - z_s(t))$.

The problem to be addressed in this chapter is to design the following mode-dependent output feedback control law,

$$\hat{u}(t) - K_i \hat{Y}_\varepsilon(t) \tag{3.6}$$

where the matrix K_i of the appropriate dimension is to be determined such that for any $r(t) = i \in S$ the resulting closed-loop system is stochastically stable and satisfies an H_∞ norm bound γ, i.e. $\mathrm{J}_\infty < 0$.

3.3 Main Results

In this section, we first analyze both the stochastic stability and H_∞ performance criterion for the system (3.1)–(3.2) with the controller (3.6). Sufficient delay-dependent stochastic stability conditions are proposed in Theorem 1. Then, we will show a procedure to design the controller gain K_i in (3.6) in terms of LMIs which guarantee the resulting closed-loop system is exponentially mean-square stable and satisfies the H_∞ performance condition.

Theorem 3.1. *Let the controller gain* K_i *and the positive scalars* $\alpha, \mathrm{h_M}, \mathrm{h_D}, \tau_\mathrm{M}$, $0 < \tau_\mathrm{D} < 1$ *be given. The Markovian jump system* (3.5) *with the control law* (3.6) *and the time-varying delays in* (3.4) *is stochastically mean square stable with an* H_∞ *performance condition* $\gamma > 0$, *if there exist some matrices* $\Omega_{1\mathrm{i}}, \Omega_{2\mathrm{i}}, \Omega_{3\mathrm{i}}$ *and positive definite matrices* $P_{1\mathrm{i}}, R_\mathrm{i}, S_\mathrm{i}, U, Z (i = 1, 2, ..., s)$ *satisfying the following LMIs*

$$\begin{bmatrix} \Sigma_1 & \mathrm{h_M}\Omega_\mathrm{i} & \mathrm{h_M}\psi_\mathrm{i}\mathrm{Z} \\ * & -\mathrm{h_M Z} & 0 \\ * & * & -\mathrm{h_M Z} \end{bmatrix} < 0 \tag{3.7a}$$

$$\sum_{\mathrm{j}=1}^{\mathrm{s}} \pi_{\mathrm{ij}} \mathrm{S_j} - (1 - \mathrm{h_D})\mathrm{R_i} \le 0, \tag{3.7b}$$

where

$$\Sigma_1 := \begin{bmatrix} \Sigma_{11} & \Sigma_{12} & \mathrm{P_{1i}A_{3i}} + \Omega_{3\mathrm{i}}^\mathrm{T} & \mathrm{e}^{\alpha\kappa}\mathrm{P_{1i}B_{2i}} & \mathrm{e}^{-\alpha\kappa}\mathrm{D}_{1\mathrm{i}}^\mathrm{T} \\ * & \Sigma_{22} & -\Omega_{3\mathrm{i}}^\mathrm{T} & 0 & \mathrm{e}^{-\alpha\kappa+\alpha\mathrm{h_M}}\mathrm{D}_{2\mathrm{i}}^\mathrm{T} \\ * & * & -(1-\tau_\mathrm{D})\mathrm{e}^{-2\alpha\kappa}\mathrm{U} & 0 & \mathrm{e}^{-\alpha\kappa}\mathrm{D}_{3\mathrm{i}}^\mathrm{T} \\ * & * & * & -\gamma^2\mathrm{I} & 0 \\ * & * & * & * & -\mathrm{I} \end{bmatrix}$$

$$\Omega_\mathrm{i} = \left[\Omega_{1\mathrm{i}}^\mathrm{T}, \Omega_{2\mathrm{i}}^\mathrm{T}, \Omega_{3\mathrm{i}}^\mathrm{T}, 0, 0\right]^\mathrm{T},$$

$$\Psi_\mathrm{i} = \left[\mathrm{A_{1i}} + \alpha\mathrm{I} - \mathrm{B_{1i}K_iC_i}, \mathrm{e}^{\alpha\mathrm{h_M}}\mathrm{A_{2i}}, \mathrm{A_{3i}}, -\mathrm{B_{2i}}, 0\right]^\mathrm{T}$$

with

$$\Sigma_{11} := \mathrm{sym}(P_{1i}(A_{1i} + \alpha I - B_{1i}K_iC_i) + \Omega_{1i}) + \sum_{j=1}^{s} \pi_{ij}P_{1j} + h_M R_i + S_i + \tau_M^2 U,$$

$$\Sigma_{12} := e^{\alpha h_M} P_{1i}A_{2i} - \Omega_{1i} + \Omega_{2i}^T,$$

$$\Sigma_{22} := -(1 - h_D)S_i - \mathrm{sym}(\Omega_{2i}).$$

Proof . Construct a stochastic LKF candidate for the system (3.5) as

$$V(\hat{e}_t, t, i) = \sum_{k=1}^{4} V_k(\hat{e}_t, t, i), \tag{3.8}$$

with

$$V_1(\hat{e}_t, t, i) = \hat{e}(t)^T P_{1i}\hat{e}(t),$$

$$V_2(\hat{e}_t, t, i) = \int_{t-h(t)}^{t} \hat{e}(\xi)^T S_i \hat{e}(\xi)\, d\xi + \int_{t-h(t)}^{t} \int_{\xi}^{t} \hat{e}(s)^T R_i\, \hat{e}(s)\, ds\, d\xi,$$

$$V_3(\hat{e}_t, t, i) = \int_{t-\tau(t)}^{t} \left[\int_{s}^{t} \hat{e}(\theta)^T d\theta\right] U \left[\int_{s}^{t} \hat{e}(\theta) d\theta\right] ds + \int_{0}^{\tau(t)} \int_{t-s}^{t} (\theta - t + s)\hat{e}(\theta)^T U \hat{e}(\theta)\, d\theta\, ds,$$

$$V_4(\hat{e}_t, t, i) = \int_{t-h_M}^{t} \int_{\xi}^{t} \dot{\hat{e}}(s)^T Z \dot{\hat{e}}(s)\, ds\, d\xi,$$

where P_{1i}, S_i, R_i, Z and U are matrices with appropriate dimensions. It is shown in [42] that the $C([-\kappa, 0]; \Re^n) \times S -$ valued process $(e_t, r(t))$ is a time homogenous strong Markov process. The weak infinitesimal operator $\pounds V(.)$ of the stochastic process $\{(\hat{e}_t, r(t)), t \geq 0\}$, acting on $V \in C(\Re^n \times \Re^+ \times S)$ at the point $\{\hat{e}_t, t, r(t) = i\}$, is given by

$$\begin{aligned}\pounds V(\hat{e}_t, t, i) &= \lim_{\Delta \to 0} + \frac{1}{\Delta}\{\varepsilon[V(\hat{e}(t+\Delta), \hat{x}_{t+\Delta}, t+\Delta, r(t+\Delta)|\hat{e}(t), \hat{e}_t, t, r(t) = i)] - V(\hat{e}, t, i)\} \\ &= V_t(\hat{e}, t, i) + \dot{\hat{e}}(t)^T V_e(\hat{e}, t, i) + \sum_{j=1}^{s} \pi_{ij} V(\hat{e}, t, j)\end{aligned} \tag{3.9}$$

where

$$V_t(\hat{e}, t, i) = \frac{\partial V(\hat{e}, t, i)}{\partial t}, V_e(\hat{e}, t, i) = \left(\frac{\partial V(\hat{e}, t, i)}{\partial \hat{e}_1}, ..., \frac{\partial V(\hat{e}, t, i)}{\partial \hat{e}_n}\right)^T.$$

Differentiating $V_1(\hat{e}_t, t, i)$ in t we obtain

$$\begin{aligned}
&\pounds V_1(\hat{e}_t, t, i) = 2\hat{e}(t)^T P_{1i}((A_{1i} + \alpha I - B_{1i}K_{1i}C_i)\hat{e}(t) \\
&+ e^{\alpha h(t)}A_{2i}\hat{e}(t - h(t)) + A_{3i}\int_{t-\tau(t)}^{t} e^{\alpha(t-s)}\hat{e}(s)ds - B_{2i}\hat{w}(t)) + \sum_{j=1}^{s}\hat{e}(t)^T\pi_{ij}P_{1j}\,\hat{e}(t)
\end{aligned} \tag{3.10}$$

Differentiating other Lyapunov terms in (3.8) give

$$\begin{aligned}
\pounds V_2(\hat{e}_t, t, i) &= h(t)\hat{e}(t)^T R_i\hat{e}(t) - (1 - \dot{h}(t))\int_{t-h(t)}^{t}\hat{e}(s)^T R_i\hat{e}(s)\,ds + \hat{e}(t)^T S_i\hat{e}(t) \\
&\quad - (1 - \dot{h}(t))\hat{e}(t - h(t))^T S_i\hat{e}(t - h(t)) + \sum_{j=1}^{s}\int_{t-h(t)}^{t}\hat{e}(\xi)^T\pi_{ij}S_j\hat{e}(\xi)d\xi \\
&\le \hat{e}(t)^T(h_M R_i + S_i)\hat{e}(t) + \int_{t-h(t)}^{t}\hat{e}(s)^T\left(\sum_{j=1}^{s}\pi_{ij}S_j - (1 - h_D)R_i\right)\hat{e}(s)\,ds \\
&\quad - (1 - h_D)\hat{e}(t - h(t))^T S_i\hat{e}(t - h(t))
\end{aligned} \tag{3.11}$$

$$\begin{aligned}
\pounds V_3(\hat{e}_t, t, i) &= -(1 - \dot{\tau}(t))\left[\int_{t-\tau(t)}^{t}\hat{e}(\theta)^T d\theta\right]U\left[\int_{t-\tau(t)}^{t}\hat{e}(\theta)d\theta\right] \\
&\quad + 2\int_{t-\tau(t)}^{t}(\theta - t + \tau(t))\hat{e}(t)^T U\hat{e}(\theta)\,d\theta \\
&\quad + \int_{0}^{\tau(t)} s\hat{e}(t)^T U\hat{e}(t)ds - \int_{0}^{\tau(t)}\int_{t-s}^{t}\hat{e}(\theta)^T U\hat{e}(\theta)\,d\theta\,ds \\
&\le \tau_M^2\hat{e}(t)^T U\hat{e}(t) - (1 - \tau_D)\left[\int_{t-\tau(t)}^{t}(e^{-\alpha\theta}\hat{e}(\theta))^T d\theta\right]U\left[\int_{t-\tau(t)}^{t}e^{-\alpha\theta}\hat{e}(\theta)\,d\theta\right]
\end{aligned} \tag{3.12}$$

and

$$\begin{aligned}\pounds V_4(\hat{x}_t, t, i) &= \dot{\hat{e}}(t)^T Z \dot{\hat{e}}(t) - \int_{t-h_M}^{t} \dot{\hat{e}}(s)^T Z \dot{\hat{e}}(s)\, ds \\ &\le \dot{\hat{e}}(t)^T Z \dot{\hat{e}}(t) - \int_{t-h(t)}^{t} \dot{\hat{e}}(s)^T Z \dot{\hat{e}}(s)\, ds \end{aligned} \tag{3.13}$$

Moreover, from the Leibniz-Newton formula, the following equation holds for any matrices $\Omega_i = \left[\Omega_{1i}^T, \Omega_{2i}^T, \Omega_{3i}^T, 0\right]^T$ with appropriate dimensions:

$$2\tilde{\chi}_e^T \Omega_i \left(\hat{e}(t) - \hat{e}(t - h(t)) - \int_{t-h(t)}^{t} \dot{\hat{e}}(s)\, ds \right) = 0 \tag{3.14}$$

From (3.7b) and (3.9)–(3.14), we obtain

$$\begin{aligned} & e^{-2\alpha t}\left(\hat{\underset{Ze}{\wedge}}(t)^T \hat{\underset{Ze}{\wedge}}(t) - \gamma^2 \widehat{w}^T(t)\widehat{w}(t) \right) + \pounds V(\hat{\underset{e}{\wedge}}, t, i) \\ & \quad \le \tilde{\chi}_e(t)^T \left(\Xi_{ei} + h_M \Omega_i Z^{-1} \Omega_i^T + h_M \Psi_i Z \Psi_i^T \right) \sim \chi_e(t) \\ & \qquad - \int_{t-h(t)}^{t} (\tilde{\chi}_e(t)^T \Omega_i + \hat{e}(s)^T Z) Z^{-1} (\tilde{\chi}_e(t)^T \Omega_i + \hat{e}(s)^T Z)^T \, ds \end{aligned} \tag{3.15}$$

where

$$\tilde{\chi}_e(t) = \left[\hat{e}(t)^T, \hat{e}(t - h(t))^T, \int_{t-\tau(t)}^{t} e^{\alpha(t-\kappa)} \hat{e}(\kappa)^T d\kappa, \widehat{w}(t)^T \right]^T \tag{3.16}$$

and

$$\begin{aligned} \tilde{\Xi}_{xi} = & \begin{bmatrix} \Sigma_{11} & \Sigma_{12} & P_{1i}A_{3i} + \Omega_{3i}^T & -P_{1i}B_{2i} \\ * & \Sigma_{22} & -\Omega_{3i}^T & 0 \\ * & * & -(1-\tau_D)e^{-2\alpha t}U & 0 \\ * & * & * & -\gamma^2 e^{-2\alpha t} I \end{bmatrix} + \begin{bmatrix} e^{-\alpha t} D_{1i}^T \\ e^{-\alpha t + \alpha h(t)} D_{2i}^T \\ e^{-\alpha t} D_{3i}^T \\ 0 \end{bmatrix}^T \\ & \times \begin{bmatrix} e^{-\alpha t} D_{1i}^T \\ e^{-\alpha t + \alpha h(t)} D_{2i}^T \\ e^{-\alpha t} D_{3i}^T \\ 0 \end{bmatrix} \end{aligned}$$

On the other hand, for a prescribed $\gamma>0$ and under zero initial conditions, one has

$$\left(\varepsilon\int_0^\infty e^{-2\alpha t}[\hat{z}_e(t)^T \hat{\Lambda}_{Ze}(t) - \gamma^2 \widehat{w}^T(t)\widehat{w}(t)]\ dt + V(\hat{\hat{e}}, t, i)\|_{t\to\infty} - V(\hat{\hat{e}}, t, i)\|_{t=0}\right)$$
$$\leq \varepsilon\left(\int_0^\infty \bar{\chi}_e(t)^T\left(\Xi_{ei} + h_M\Omega_i Z^{-1}\Omega_i^T + h_M\psi_i Z\psi_i^T\right)\tilde{\chi}_e(t)dt\right) \tag{3.17}$$

Then, the condition

$$\tilde{\Xi}_{ei} + h_M\Omega_i Z^{-1}\Omega_i{}^T + h_M\psi_i Z\psi_i^T < 0 \tag{3.18}$$

means that the H_∞ performance criteria is satisfied. Furthermore, by the Schur complement lemma, the condition (3.7a) is obtained.

It is noting that the condition (3.15) for $w(t) = 0$ implies $\varepsilon\{£V(\hat{e},t,i)\} < 0$. Then, we have

$$\varepsilon\{\ £V(\hat{e}, t, i)\} \leq -\sigma_1\varepsilon\{\hat{e}(t)^T\hat{e}(t)\} \tag{3.19}$$

where $\sigma_1 = \min\{\lambda_{\min}(-\tilde{\Xi}_{e1}), i \in S\}$, then $\sigma_1 > 0$. By Dynkin's formula, we have

$$\varepsilon\left\{\int_0^\tau \hat{e}(s)^T\hat{e}(s)\ ds\right\} \leq \sigma_1^{-1}V(\phi(0), 0, r_0) \tag{3.20}$$

Moreover, from LKF (3.8) the following condition holds

$$\varepsilon\{V(\hat{e}, t, i)\} \geq \sigma_2\varepsilon\left\{\hat{e}(t)^T\hat{e}(t)\right\} \tag{3.21}$$

where $\sigma_2 = \min\{\lambda_{\min}(P_{1i}), i \in S\}$. From (3.19) to (3.21), we obtain

$$\varepsilon\left\{\hat{e}(t)^T\hat{e}(t)\right\} \leq -\sigma_1\sigma_2^{-1}\varepsilon\left\{\int_0^\tau \hat{e}(s)^T\hat{e}(s)\ ds\right\} + \sigma_2^{-1}V(\phi(0), 0, r_0).$$

Hence

$$\lim_{t\to x}\varepsilon\left\{\int_0^t \hat{e}(s)^T\hat{e}(s)\ ds\right\} \leq \sigma_1^{-1}\ V(\phi(0),\ 0, r_0) \tag{3.22}$$

Moreover, from the Chebyshev integral inequality it can be shown that

$$\lim_{t-x}\varepsilon\left\{\int_0^\tau \hat{e}(s)^T\hat{e}(s)\ ds\right\} \leq 2\sigma_1^{-1}\ e^{-\alpha t}V(\phi(0),\ 0, r_0) \tag{3.23}$$

which indicates that, from Definition 3.1, the system in (3.5) with Markovian switching parameters is globally exponentially stable in the mean square sense and has the exponential decay rate α. This completes the proof.

Up to now, the H_∞ performance-based stability analysis problem of the Markovian jump system with time-varying delays has been solved. Now we are in the position to solve the synchronization problem of the system (3.1)–(3.2). In the following theorem, we can obtain a mode-dependent output feedback control law in the form of (3.6).

Lemma 3.1. [43] *For given positive-definite matrices* Γ *and* θ, *the nonlinear matrix inequality* $K^T\theta^{-1}K < \Gamma^{-1}$ *is satisfied if the following LMI holds*

$$\begin{bmatrix} -2X + \Gamma & W^T \\ * & -\theta \end{bmatrix} < 0$$

With $W = KX$:.

Theorem 3.2. *Consider the Markovian jump system* (3.1)–(3.2) *with time-varying delays* (3.3). *For given positive scalars* $\alpha, h_M, h_D, \tau_M, 0 < \tau_D < 1$ *and* γ, *there exits a mode-dependent output feedback control in the form of (3.6) such that the resulting closed-loop system is stochastically exponentially stable and satisfies an* H_∞ *performance criterion in the sense of Definitions* 3.1 *and* 3.2, *if there exist matrices* $Y_{C_i}, \Gamma_i, \widehat{\Omega}_{1i}, \widehat{\Omega}_{2i}, \widehat{\Omega}_{3i}$, *and positive-definite matrices* $\bar{P}_{Q_i}, \bar{P}_{C_i}, \bar{Z}, \widehat{S}_i, \widehat{R}_i, \widehat{U}$. $(i = 1, 2, ..., s), ..,$ *satisfying the following LMIs*

$$\begin{bmatrix} \tilde{\Sigma}_1 & h_M\widehat{\Omega}_i & h_M\tilde{\psi}_i \\ * & -h_M\Gamma_i & 0 \\ * & * & -h_M\bar{Z} \end{bmatrix} < 0 \tag{3.24a}$$

$$\sum_{j=1}^{s} \pi_{ij}\hat{S}_j - (1 - h_D)\hat{R}_i \le 0, \tag{3.24b}$$

$$\begin{bmatrix} -2Q_i\bar{P}_{Q_i}Q_i^T - 2C_i^{\perp}\bar{P}_{C_i}C_i^{\perp T} + \Gamma_i & I \\ * & -\bar{Z} \end{bmatrix} < 0 \tag{3.24c}$$

where $\hat{\Omega}_i = [\hat{\Omega}_{1i}^T, \hat{\Omega}_{2i}^T, \hat{\Omega}_{3i}^T, 0]^T$, and

$$\begin{aligned} \widehat{\Psi} = [&(A_{1i} + \alpha 1)(Q_i\bar{P}_{Q_i}Q_i^T + C_i^{\perp}\bar{P}_{C_i}C_i^{\perp T}) - B_{1i}Y_{C_i}C_i^{\perp T}, e^{\alpha h_M}A_{2i}(Q_i\bar{P}_{Q_i}Q_i^T \\ &+ C_i^{\perp}\bar{P}_{C_i}C_i^{\perp T}), A_{3i}(Q_i\bar{P}_{Q_i}Q_i^T + C_i^{\perp}\bar{P}_{C_i}C_i^{\perp T}), -B_{2i}(Q_i\bar{P}_{Q_i}Q_i^T \\ &+ C_i^{\perp}\bar{P}_{C_i}C_i^{\perp T})]^T \end{aligned}$$

and

$$\tilde{\Sigma}_1 := \begin{bmatrix} \tilde{\Sigma}_{11} & \tilde{\Sigma}_{12} & \tilde{\Sigma}_{13} & -\mathrm{e}^{\alpha\kappa}\mathrm{B}_{2\mathrm{i}} & \tilde{\Sigma}_{15} \\ * & -(1-\mathrm{h_D})\,\hat{\mathrm{S}}_{\mathrm{i}} - \mathrm{sym}(\hat{\Omega}_{2\mathrm{i}}) & -\hat{\Omega}_{3\mathrm{i}}^{\mathrm{T}} & 0 & \tilde{\Sigma}_{25} \\ * & * & -(1-\tau_{\mathrm{D}})\mathrm{e}^{-2\alpha\kappa}\hat{\mathrm{U}} & 0 & \tilde{\Sigma}_{35} \\ * & * & * & -\gamma^2\,\mathrm{I} & 0 \\ * & * & * & * & -\mathrm{I} \end{bmatrix}$$

with

$$\begin{aligned}
\tilde{\Sigma}_{11} :&= sym((A_{1i}+\alpha I)Q_i\bar{P}_{Q_i}Q_i^T + (A_{1i}+\alpha I)C_i^{\perp}\bar{P}_{C_i}C_i^{\perp T} + \widehat{\Omega}_{1i}) \\
&+sym(-B_{1i}Y_{C_i}C_i^{\perp T}) + \sum_{j=1}^{s}\pi_{ij}(Q_j\bar{P}_{Q_j}Q_j^T + C_j^{\perp}\bar{P}_{C_j}C_j^{\perp T}) + h_M\widehat{R}_i + \widehat{S}_i + \tau_M^2\widehat{U}, \\
\tilde{\Sigma}_{12} &:= e^{\alpha h_M}A_{2i}Q_i\bar{P}_{Q_i}Q_i^T + e^{\alpha h_M}A_{2i}C_i^{\perp}\bar{P}_C C_i^{\perp T} - \widehat{\Omega}_{1i} + \widehat{\Omega}_{2i}^T, \\
\tilde{\Sigma}_{13} &:= A_{3i}Q_i\bar{P}_{Q_i}Q_i^T + A_{3i}C_i^{\perp}\bar{P}_{C_i}C_i^{\perp T} + \widehat{\Omega}_{3i}, \\
\tilde{\Sigma}_{15} &:= e^{-\alpha\kappa}Q_i\bar{P}_{Q_i}Q_i^T D_{1i}^T + e^{-\alpha\kappa}C_i^{\perp}\bar{P}_{C_i}C_i^{\perp T}D_{1i}^T, \\
\tilde{\Sigma}_{25} &:= e^{-\alpha\kappa+\alpha h}{}_M Q_i\bar{P}_{Q_i}Q_i^T D_{2i}^T + e^{-\alpha\kappa+\alpha h}{}_M C_i^{\perp}\bar{P}_{C_i}C_i^{\perp T}D_{2i}^T, \\
\tilde{\Sigma}_{35} &:= e^{-\alpha\kappa}Q_i\bar{P}_{Q_i}Q_i^T D_{3i}^T + e^{-\alpha\kappa}C_i^{\perp}\bar{P}_{C_i}C_i^{\perp T}D_{3i}^T,
\end{aligned}$$

The desired control gains in (3.6) are given by $K_i = Y_{C_i}\bar{P}_{C_i}^{-1}$.

Proof. Let $\xi = diag\{\bar{P}_{1i}, \bar{P}_{1i}, \bar{P}_{1i}, I, \bar{P}_{1i}, I\}$, where $\bar{P}_{1i} := P_{1i}^{-1}$. By performing a congruence transformation ξ, to both sides of (3.7a) and considering $Y_i := K_iC_i\bar{P}_{1i}$, $\widehat{U} := \overline{P}_{1i}U\overline{P}_{1i}$, $\widehat{R}_i := \overline{P}_{1i}R_i\overline{P}_{1i}$, $\widehat{S}_i := \overline{P}_{1i}S_i\overline{P}_{1i}$ and $\widehat{P}_{1j} := \overline{P}_{1i}P_{1j}\overline{P}_{1i}$ ($j = 1, ..., n$; $j \neq i$), we obtain

$$\begin{bmatrix} \widehat{\Sigma}_1 & \mathrm{h_M}\widehat{\Omega}_{\mathrm{i}} & \mathrm{h_M}\widehat{\Psi}_{\mathrm{i}} \\ * & -\mathrm{h_M}\overline{P}_{1i}\overline{Z}\overline{P}_{1i} & 0 \\ * & * & -\mathrm{h_M}\overline{\mathrm{Z}} \end{bmatrix} < 0 \tag{3.25}$$

where

$$\hat{\Omega}_i = \left[\hat{\Omega}_{1i}^{\mathrm{T}}, \hat{\Omega}_{2i}^{\mathrm{T}}, \hat{\Omega}_{3i}^{\mathrm{T}}, 0, 0\right]^{\mathrm{T}},$$

$$\widehat{\Psi}_i = [(\mathrm{A_{1i}} + \alpha\mathrm{I})\overline{P}_{1i} - \mathrm{B_{1i}Y_i}, \mathrm{e}^{\alpha\mathrm{h_M}}\mathrm{A_{2i}}\overline{P}_{1i}, \mathrm{A_{3i}}\overline{P}_{1i}, -\mathrm{B_{2i}}\overline{P}_{1i}, 0]^{\mathrm{T}},$$

$$\hat{\Sigma}_1 := \begin{bmatrix} \Sigma_{11} & \mathrm{e}^{\alpha\mathrm{h_M}}\mathrm{A_{2i}}\bar{\mathrm{P}}_{1i} & \mathrm{A_{3i}}\bar{\mathrm{P}}_{1i} & -e^{\alpha\kappa}\mathrm{P}_{1i}\mathrm{B}_{2i} & \mathrm{e}^{-\alpha\kappa}\bar{\mathrm{P}}_{1i}\mathrm{D}_{1\mathrm{i}}^{\mathrm{T}} \\ * & -(1-h_D)\,\hat{\mathrm{S}}_i & 0 & 0 & e^{-\alpha\kappa+\alpha\,\mathrm{h_M}}\bar{\mathrm{P}}_{1i}\mathrm{D}_{2\mathrm{i}}^{\mathrm{T}} \\ * & * & -(1-\tau_D)e^{-2\alpha t}\hat{\mathrm{U}} & 0 & e^{-\alpha\kappa}\bar{P}_{1i}\mathrm{D}_{3\mathrm{i}}^{\mathrm{T}} \\ * & * & * & -\gamma^2\,\mathrm{I} & 0 \\ * & * & * & * & -\mathrm{I} \end{bmatrix}$$

with $\hat{\Omega}_{ji} = \bar{P}_{1i}\Omega_{ji}\bar{P}_{1i}$, $\overline{Z} := Z^{-1}$ and

$$\hat{\Sigma}_{11} := \mathrm{sym}((A_{1i} + \alpha I)\bar{P}_{1i} - B_{1i}Y_i + \hat{\Omega}_{1i}) + \sum_{j=1}^{s} \pi_{ij}\hat{P}_{1j} + h_M\hat{R}_i + \hat{S}_i + \tau_M^2\hat{U}$$

Let us consider the C_i Moore–Penrose pseudo-inverse $C_i^{\perp} = C_i^T(C_iC_i^T)^{-1}$ and an $n \times (n-p)$ matrix Q_i whose columns are a basis of $Ker(C_i)$. According to [44], we define the transformations

$$\overline{P}_{1i} = Q_i\overline{P}_{Q_i}Q_i^T + C_i^{\perp}\overline{P}_{C_i}C_i^{\perp T}, \tag{3.26a}$$

$$Y_i = Y_{C_i}C_i^{\perp T}, \tag{3.26b}$$

where $\overline{P}_{Q_i}, \overline{P}_{C_i}$ are symmetric matrices with respective dimensions $(n-p)\times(n-p)$, $p \times p$; and Y_{C_i} is an arbitrary $m \times p$ matrix. The LMI (3.25) can be expressed in terms of the new variable matrices $\overline{P}_{Q_i}\overline{P}_{C_i}Y_{C_i}$. Using $C_iQ_i = 0, C_iC_i^{\perp} = I$, left-multiplying $\overline{P}_{1i}$ by C_i, we obtain $C_i\overline{P}_{1i} = C_i(Q_i\overline{P}_{Q_i}Q_i^T + C_i^{\perp}\overline{P}_{C_i}C_i^{\perp T}) = \overline{P}_{C_i}C_i^{\perp T}$ and $Y_i = Y_{C_i}C_i^{\perp T}$ $= Y_{C_i}\overline{P}_{Ci}^{-1}\overline{P}_{C_i}C_i^{\perp T} = Y_{C_i}\overline{P}_{C_i}^{-1}C_i\overline{P}_{1i}$. Then, according to the definition $Y_i := K_iC_i\overline{P}_{1i}$, the control gain K_i is obtained by $K_i = Y_{C_i}\overline{P}_{C_i}^{-1}$. Furthermore, by Lemma 3.1, it is easily seen that the matrix inequality $\overline{P}_{1i}\overline{ZP}_{1i} > \Gamma_i$ (or $P_{1i}ZP_{1i} < \Gamma_1^{-1}$) holds if the LMI (3.24c) can be satisfied. Then, the matrix inequality (3.25) yields the LMI (3.24a). This completes the proof.

3.4 Simulation Results

Consider a Markovian jump system (3.1) with two Markovian switching modes and the following state-space matrices

Mode 1

$$A_1(1) = \begin{bmatrix} 1.01 & 2.01 \\ 0.99 & -0.10 \end{bmatrix}; \quad A_2(1) = \begin{bmatrix} 0.11 & 0 \\ 0.11 & -0.11 \end{bmatrix};$$

$$A_3(1) = \begin{bmatrix} -0.8 & 0 \\ 0.1 & -1.1 \end{bmatrix};$$

$$B_1(1) = \begin{bmatrix} 2 \\ 1 \end{bmatrix}; \quad B_2(1) = \begin{bmatrix} 1 \\ 0.2 \end{bmatrix};$$

$$D_1(1) = [\,1 \quad 0.2\,]; \quad D_2(1) = [\,-1 \quad 0.2\,];$$

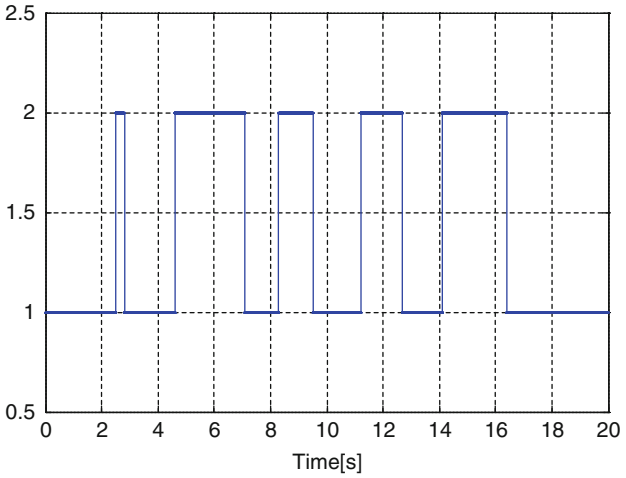

Fig. 3.1 Random jumping mode

Mode 2

$$A_1(2) = \begin{bmatrix} -1.01 & 0.59 \\ 1.09 & -2.01 \end{bmatrix}; A_2(2) = \begin{bmatrix} 0.11 & 0.10 \\ 0.20 & -0.21 \end{bmatrix}; A_3(2) = \begin{bmatrix} -1.1 & 0.1 \\ 0 & 0.5 \end{bmatrix};$$

$$B_1(2) = \begin{bmatrix} 1 \\ 2 \end{bmatrix}; B_2(2) = \begin{bmatrix} 1 \\ 1 \end{bmatrix}; D_1(2) = [0.1 \quad 1]; D_2(2) = [0.1 \quad -0.1]$$

with $h_M = 2.5;\ \ h_D = 1.15; \tau_M = 0.8; \tau_D = 0.3$.

The following transition matrix is considered.

$$\pi = \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix}$$

A realization of the jumping mode is plotted in Fig. 3.1, where the initial mode is assumed to be $r_0 = 1$. For simulation purpose, we simply choose the disturbance as $w(t) = 1/(2 + 0.5t^2)$ and the initial condition $x(0) = [2 - 1]^T$.

We choose the output matrices as

$$C(1) = \begin{bmatrix} -0.8 & 0.1 \\ 0.3 & 0.5 \end{bmatrix};$$

$$C(2) = \begin{bmatrix} 0.1 & 0.4 \\ 0.7 & -0.2 \end{bmatrix}$$

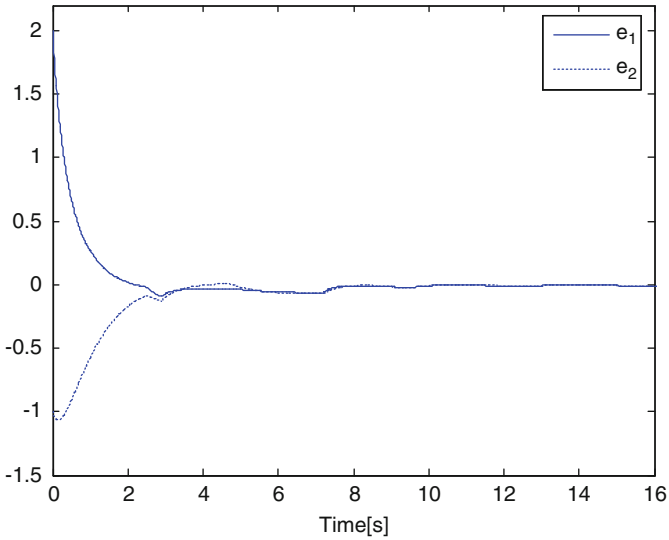

Fig. 3.2 Synchronization errors

It is clear that the matrices Q(1) and Q(2) should be zeros. Under $\gamma = 0.9485$, after solving the LMIs (3.24), the LMI variables are obtained as follows:

$$\overline{P}_{Q_1} = \begin{bmatrix} 4.3535 & -1.4945 \\ -1.4945 & 2.2963 \end{bmatrix};$$

$$\overline{P}_{C_1} = \begin{bmatrix} 1.4261 & -0.1222 \\ -0.1222 & 0.3085 \end{bmatrix}$$

$$\overline{P}_{Q_2} = \begin{bmatrix} 4.0501 & 1.9884 \\ 1.9884 & 7.7804 \end{bmatrix};$$

$$\overline{P}_{C_2} = \begin{bmatrix} 0.5392 & 0.0126 \\ 0.0126 & 1.6597 \end{bmatrix}$$

$$\mathrm{Y}_{C_1} = [-1.3591 \quad 1.0997];$$

$$\mathrm{Y}_{C_2} = [-0.7287 \quad -0.2845]$$

The control gains are calculated as

$$\mathrm{K}(1) = [-0.6703 \ 3.2990];$$

$$\mathrm{K}(2) = [-1.3478 - 0.1612].$$

Responses of the two synchronization error signals of the closed-loop system are depicted in Fig. 3.2. The exponential convergence of the states can be recognized in the figure.

3.5 Conclusion

In this chapter, an output feedback control proach was proposed for exponential H_∞ synchronization problem of master–slave systems with both discrete and distributed time-varying delays and Markovian switching parameters. Using an appropriate Lyapunov–Krasovskii functional, some delay-dependent sufficient conditions and a synchronization law which include the master–slave parameters were established for designing a mode-dependent output feedback control law in terms of linear matrix inequalities. The controller guarantees the H_∞ synchronization of the two coupled master and slave systems regardless of their initial states. A numerical example was given to show the effectiveness of the method.

References

1. Boukas, E.K., Benzaouia, A.: Stability of discrete-time linear systems with Markovian jumping parameters and constrained control. IEEE Trans. Automat. Contr. **47**(3), 516–521 (2002)
2. Boukas, E.K., Zhang, Q., Yin, G.: Robust production and maintenance planning in stochastic manufacturing systems. IEEE Trans. Automat. Contr. **40**(6), 1098–1102 (1995)
3. Cao, Y., Lam, J.: Stochastic stabilizability and H_∞ control for discrete-time jump linear systems with time delay. J Franklin Institute **336**(8), 1263–1281 (1999)
4. Shi, P., Boukas, E.K.: H_∞ control for Markovian jumping linear systems with parametric uncertainty. J. Optim. Theor. Appl. **95**(1), 75–99 (1997)
5. Mariton, M.: Jump Linear Systems in Automatic Control. Marcel Dekker Inc, New York (1990)
6. Mahmoud, M.S., Shi, P.: Robust control for Markovian jump linear discrete-time systems with unknown nonlinearities. IEEE Trans. Circ. Syst. **I 49**(4), 538–542 (2002)
7. Malmoud, M.S., Shi, P., Shi, Y.: H_∞ and robust control of interconnected systems with Markovian jump parameters. J. Discrete Continuous Dyn. Syst. **B 5**(2), 365–384 (2005)
8. Mahmoud, M.S., Shi, P.: Robust stability, stabilization and H_∞ control of time-delay Systems with Markovian jump parameters. J. Robust Nonlinear Contr. **13**(8), 755–784 (2003)
9. Mahmoud, M.S., Shi, P.: Output feedback stabilization and disturbance attenuation of time-delay jumping systems. IMA J. Math. Contr. Inform. **20**(2), 179–199 (2003)
10. Zhang, L., Boukas, E.: Stability and stabilization of Markovian jump linear systems with partly unknown transition probability. Automatica **45**(2), 463–468 (2009)
11. Zhang, L., Lam, J.: Necessary and sufficient conditions for analysis and synthesis of Markov jump linear systems with incomplete transition descriptions. IEEE Trans. Automat. Contr. **55**(7), 1695–1701 (2010)
12. Hale, J., Verduyn Lunel, S.M.: Introduction to Functional Differential Equations. Springer, New York, NY (1993)

13. Karimi, H.R., Zapateiro, M., Luo, N.: A linear matrix inequality approach to robust fault detection filter design of linear systems with mixed time-varying delays and nonlinear perturbations. J. Franklin Inst. **347**, 957–973 (2010)
14. Wang, Z., Lam, J., Liu, X.H.: Nonlinear filtering for state delayed systems with Markovian switching. IEEE Trans. Signal Process. **51**, 2321–2328 (2003)
15. Xu, S., Chen, T., Lam, J.: Robust H_∞ filtering for uncertain Markovian jump systems with mode-dependent time delays. IEEE Trans. Automat. Contr. **48**(5), 900–907 (2003)
16. Wang, Z., Lam, J., Liu, X.: Exponential filtering for uncertain Markovian jump time-delay systems with nonlinear disturbances. IEEE Trans. Circ. Syst. II Exp. Briefs **51**(5), 262–268 (2004)
17. Mahmoud, M.S., Shi, P.: Robust Kalman filtering for continuous time-lag systems with Markovian jump parameters. IEEE Trans. Circ. Syst. **I 50**, 98–105 (2003)
18. Yang, H., Xia, Y., Shi, P.: Observer-based sliding mode control for a class of discrete systems via delta operator approach. J. Franklin Inst. **347**(7), 1199–1213 (2010)
19. Karimi, H.R.: Robust delay-dependent H_∞ control of uncertain time-delay systems with mixed neutral, discrete and distributed time-delays and Markovian switching parameters. IEEE Trans. Circ. Syst. **I 58**(8), 1910–1923 (2011)
20. Chen, D., Zhang, W.: Sliding mode control of uncertain neutral stochastic systems with multiple delays. Math. Probl. Eng. 2008, Article ID 761342, 9 pages
21. Xia, Y., Jia, Y.: Robust sliding-mode control for uncertain time-delay systems: An LMI approach. IEEE Trans. Automat. Contr. **48**, 1086–1092 (2003)
22. Shi, P., Xia, Y., Liu, G.P., Rees, D.: On designing of sliding-mode control for stochastic jump systems. IEEE Trans. Automat. Contr. **51**, 97–103 (2006)
23. Zhang, L., Boukas, E.: Mode-dependent H_∞ filtering for discrete-time Markovian jump linear systems with partly unknown transition probability. Automatica **45**(6), 1462–1467 (2009)
24. Zhang, L.: H_∞ estimation for piecewise Homogeneous Markov jump linear systems. Automatica **45**(11), 2570–2576 (2009)
25. Lin, Z., Xia, Y., Shi, P., Wu, H.: Robust sliding mode control for uncertain linear discrete systems independent of time-delay. Int. J. Innovat. Comput. Inform. Contr. **7**(2), 869–881 (2011)
26. Niu, Y., Ho, D.W.C.: Stabilization of discrete-time stochastic systems via sliding mode technique. J. Franklin Instit. (in Press). doi:10.1016/j.jfranklin.2011.06.005
27. Mahmoud, M.S., Shi, P.: Methodologies for Control of Jumping Time-Delay Systems. Kluwer Academic Publishers, Amsterdam (2003)
28. Wu, L., Shi, P., Gao, H.: State estimation and sliding mode control of Markovian jump singular systems, IEEE Trans. Automat. Contr. **55**(5), 1213–1219 (2010)
29. Wu, L., Ho, D.W.C.: Sliding mode control of singular stochastic hybrid systems, Automatica **46**, 779–783 (2010)
30. Niu, Y., Ho, D.W.C., Wang, X.: Sliding mode control for Itô stochastic systems with Markovian switching, Automatica **43**, 1784–1790 (2007)
31. Pecora, L.M., Carroll, T.L.: Synchronization in chaotic systems. Phys. Rev. Lett. **64**, 821–824 (1990)
32. Femat, R., Alvarez-Ramírez, J., Fernández-Anaya, G.: Adaptive synchronization of high-order chaotic systems: a feedback with low-order parametrization. Phys. D Nonl. Phenomena **139** (3–4), 231–246 (2000)
33. Liao, T.L., Tsai, S.H.: Adaptive synchronization of chaotic systems and its application to secure communication. Chaos Solitons Fractals **11**(9), 1387–1396 (2000)
34. Feki, M.: An adaptive chaos synchronization scheme applied to secure communication. Chaos Solitons Fractals **18**, 141–148 (2003)
35. Wang, Y.W., Wen, C., Soh, Y.C., Xiao, J.W.: Adaptive control and synchronization for a class of nonlinear chaotic systems using partial system states. Phys. Lett. **A 351**(1–2), 79–84 (2006)
36. Gao, H., Lam, J., Chen, G.: New criteria for synchronization stability of general complex dynamical networks with coupling delays. Phys. Lett. **A 360**(2), 263–273 (2006)

37. Karimi, H.R., Maass, P.: Delay-range-dependent exponential H_∞ synchronization of a class of delayed neural networks. Chaos Solitons Fractals **41**(3), 1125–1135 (2009)
38. Wen, G., Wang, Q.G., Lin, C., Han, X., Li, G.: Synthesis for robust synchronization of chaotic systems under output feedback control with multiple random delays. Chaos Solitons Fractals **29**(5), 1142–1146 (2006)
39. Cao, J., Lu, J.: Adaptive synchronization of neural networks with or without time-varying delay. Chaos 16, 013133 (2006)
40. Karimi, H.R., Gao, H.: New delay-dependent exponential H_∞ synchronization for uncertain neural networks with mixed time delays. IEEE Trans. Syst. Man Cybernet. **B 40**(1), 173–185 (2010)
41. Wang, L., Cao, J.: Global robust point dissipativity of interval neural networks with mixed time-varying delays. Nonlinear Dynam. **55**(1–2), 169–178 (2009)
42. Kolmanovskii, V., Koroleva, N., Maizenberg, T., Mao, X., Matasov, A.: Neutral stochastic differential delay equations with Markovian switching. Stoch. Anal. Appl. **21**, 819–847 (2003)
43. Karimi, H.R.: Observer-based mixed $_s$ control design for linear systems with time-varying delays: An LMI approach. Int. J. Contr. Automat. Syst. **6**(1), 1–14 (2008)
44. Zečević, A.I., Šiljak, D.D.: Design of robust static output feedback for large-scale systems. IEEE Trans. Automat. Contr. **49**(11), 2040–2044 (2004)

Part 2
Nonlinear Dynamics and Chaos in Electronics

Chapter 4
Chaotic Behavior of Transistor Circuits

M.P. Hanias, H.E. Nistazakis, and G.S. Tombras

4.1 Introduction

The study of simple circuits which are composed by externally triggered single transistors can give significant results for the possibility of creation controlled chaotic outputs [1–3]. This result is very important, especially in telecommunication systems where the requirement for safe signal transmission is imperative. In this work, we present the chaotic behavior of three forms of such circuits and the corresponding theoretical methodology. Thus, we investigate the operation of an externally triggered single transistor chaotic circuit powered by a sinusoidal voltage [1–3]. Moreover, the correlation between the emitter's and collector's chaotic voltage time series is studied by applying the nearest neighbor algorithm, for the prediction of the emitter's voltage time series from the reconstruction of its attractor, which is based on estimating the number of the collector's nearest neighbor voltage time series for specific time interval.

The next step is the implementation of an experimental externally triggered chaotic circuit [1–3], with a bipolar junction transistor operating in its reverse active region is studied to investigate it for possible control features in its output phase portraits. Nonlinear time series modeling techniques applied to analyze the circuit's output voltage oscillations and reveal the presence of chaos, which is achieved by controlling the amplitude of the applied input signal. The phase space, that describes the behavior evolution of a nonlinear system, is reconstructed using the delay embedding theorem suggested by Takens. The embedding dimension and the largest Lyapunov exponent are estimated using the false nearest-neighbor algorithm. Moreover, the phase space embedding based weight predictor algorithm, is

M.P. Hanias (✉) • H.E.Nistazakis • G.S. Tombras
Faculty of Physics, Department of Electronics, Computers, Telecommunications and Control, National and Kapodistrian University of Athens, Athens, 15784 Greece
e-mail: mhanias@gmail.com; enistaz@phys.uoa.gr; gtombras@phys.uoa.gr

S. Banerjee et al. (eds.), *Applications of Chaos and Nonlinear Dynamics in Science and Engineering - Vol. 2*, Understanding Complex Systems,
DOI 10.1007/978-3-642-29329-0_4,

employed to make a short-term prediction of the chaotic time series for which the system's governing equations are unknown.

Next, in the study of simple chaotic circuits, we consider the case of an optoelectronic chaotic circuit which is based on an optocoupler device and can be used as a controlled optoelectronic chaotic signal generator [1–3]. The complete circuit is simple and its software simulated operation demonstrates how chaos can be generated and then how can be controlled by varying specific circuit parameters. The minimum embedding dimension is estimating using the method of false nearest neighbors. With the same methodology, we examine, [1–3], the chaotic behavior of a single electron transistor circuit where, a single electron transistor's equivalent circuit is presented.

This chapter is organized as follow. In section two we investigate the externally triggered single transistor chaotic circuit, in section three we present the externally triggered experimental chaotic circuit with a bipolar junction transistor operating in its reverse active region, while in sections four and five we study an optoelectronic chaotic circuit which is based on an optocoupler device and can be used as a controlled optoelectronic chaotic signal generator and a single electron transistor circuit respectively.

4.2 An Externally Triggered Single Transistor Chaotic Circuit

The complete layout of the considered circuit is shown in Fig. 4.1a. It consists of a basic common emitter configuration of a BC107BP npn-type BJT transistor with an emitter degeneration resistor $R_1 = 3$ KΩ and a collector resistor $R_2 = 30$ Ω in series. The circuit is driven by an input sinusoidal voltage with amplitude V_1 as applied through an inductor $L = 75$ μH directly to the transistor base and is powered supplied by a sinusoidal voltage υ_2 with amplitude V_2 connected to the transistor collector through R_2. Its operation is examined by means of the MultiSim circuit simulation environment, as illustrated in Fig. 4.1b, and by monitoring voltage υ_y across the emitter resistor R_1 and voltage υ_x across the collector resistor R_2.

It is clear that both voltages υ_x and υ_y depend on the collector current iC(t) which, under certain conditions, turns to be an important circuit parameter, since it will become chaotic when the circuit exhibits chaotic operation. For example, the initial RLT circuit of [4] exhibits chaotic operation although triggered by a sinusoidal (and definitely not chaotic) input voltage. This chaotic operation has been explained by the chaotic nature of the collector current as a result of the biasing state of both base-emitter and collector-base junctions that could have lead the transistor to operate in its reverse active region [4]. Following that and considering the incorporated RLT circuit as shown in Fig. 4.1, it is reasonable to expect that both the amplitude and frequency values of input signal $\upsilon_1 = V_1 \sin(\omega_1 t)$ as well as the supply voltage $\upsilon_2 = V_2 \sin(\omega_2 t)$ may strongly affect its operation and most probably, the presence of chaos. This can be seen in Fig. 4.2, where depicts the obtained chaotic time series of the output signal $\upsilon_y(t) = iC(t)R_1$ for input signal amplitude $V_1 = 13$ Volts and frequency $f_1 = 1$ KHz and $V_2 = 12$ Volts and frequency $f_2 = 900$ Hz.

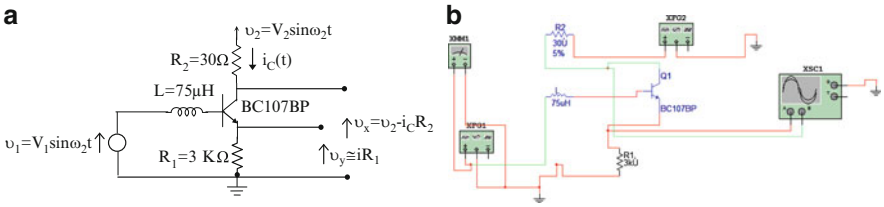

Fig. 4.1 (**a**) The considered RLT circuit, and (**b**) its simulation environment

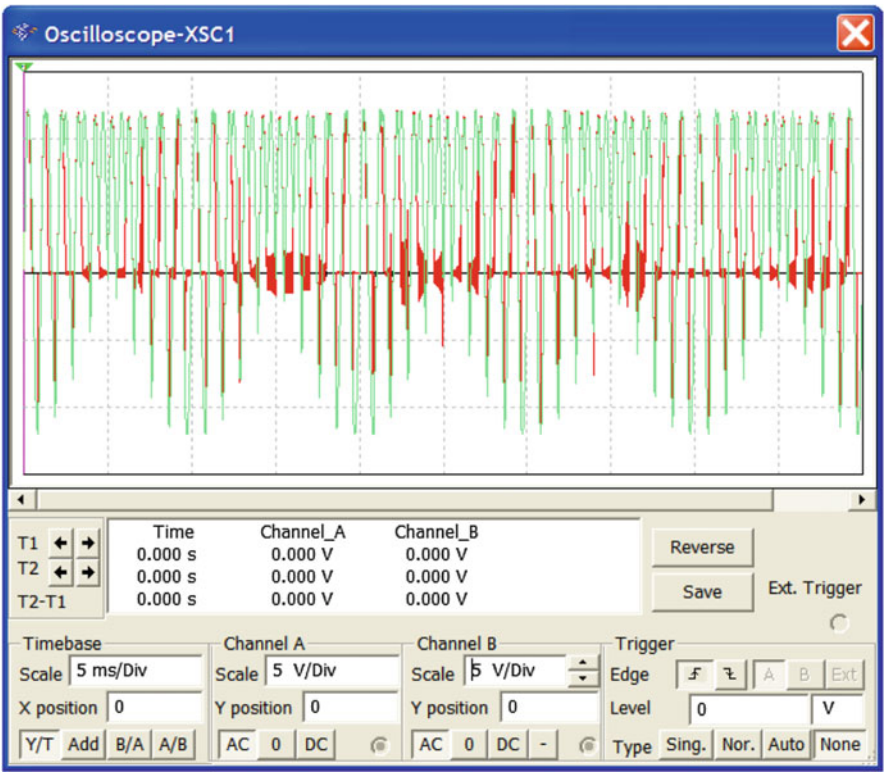

Fig. 4.2 Output chaotic signal $\upsilon_y = \upsilon_y(t)$ (*red line*) across R_1 and $\upsilon_x = \upsilon_x(t)$ across R_2 for the RLT circuit of Fig. 4.1b. With $f_1 = 1$ KHz and $f_2 = 900$ Hz both are chaotic

Keeping fixed the values of R_1, R_2 and L we then examine the influence of the AC supply voltage $\upsilon_2 = V_2\sin(\omega_2 t)$ relative to its input signal $\upsilon_1 = V_1\sin(\omega_1 t)$ towards enhancement or elimination of chaotic state. Indeed, for $V_1 = 13$ Volts, $f_1 = \omega_1/2\pi = 1$ KHz and $V_2 = 12$ Volts, $f_2 = 900$ Hz the time series of voltage υ_y (red or dark grey line in Fig. 4.2) and the time series of voltage υ_x (green or light grey line in Fig. 4.2) will become chaotic. By increasing V_2 to 13 Volts and 14 Volts the situation will not change and both voltages will remain chaotic. However, for

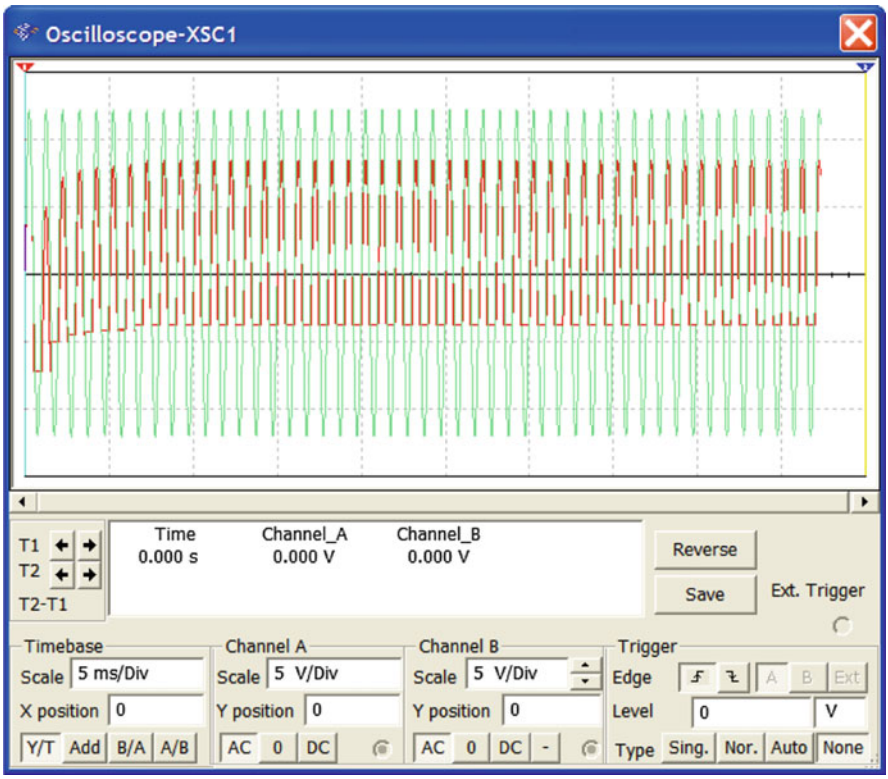

Fig. 4.3 Output chaotic signal $\upsilon_y = \upsilon_y(t)$ (*red or dark grey line*) across emitter resistor R_1 and $\upsilon_x = \upsilon_x(t)$ across the collector resistor R_2 for the RLT circuit of Fig. 4.1b. For $f_1 = f_2 = 1$ KHz both are periodic

$V_2 = 12$ Volts and $f_1 = f_2 = 1$ KHz both time series $\upsilon_x = \upsilon_x(t)$, $\upsilon_y = \upsilon_y(t)$ will retain their periodic waveform as shown in Fig. 4.3, while for $f_1 = 1$ KHz and $f_2 = 999$ Hz the signal $\upsilon_y = \upsilon_y(t)$ (red line) across emitter resistor R_1 will become chaotic despite the fact that the signal $\upsilon_x = \upsilon_x(t)$ across collector resistor R_2 will remain periodic, as shown in Fig. 4.4. Moreover it is interesting to see that as Fig. 4.5 shows, the lower limit of f_2 for which the $\upsilon_x = \upsilon_x(t)$ time series is "weakly" chaotic is $f_2 = 995$ Hz. Hence we conclude that f_2 strongly affects both the signal across emitter resistor and the signal across collector resistor.

4.2.1 Nearest Neighbor Algorithm for Cross Prediction

In order to examine the correlation between the chaotic signals across the emitter resistor and collector resistor and the manner they affect each other, we will try to cross-predict $\upsilon_y = \upsilon_y(t)$ signal from $\upsilon_x = \upsilon_x(t)$. This can be done using the nearest neighbor algorithm (NN) [5–10]. The main idea of this algorithm is that each time series copies its own behavior along the time axis. In other words, parts of past

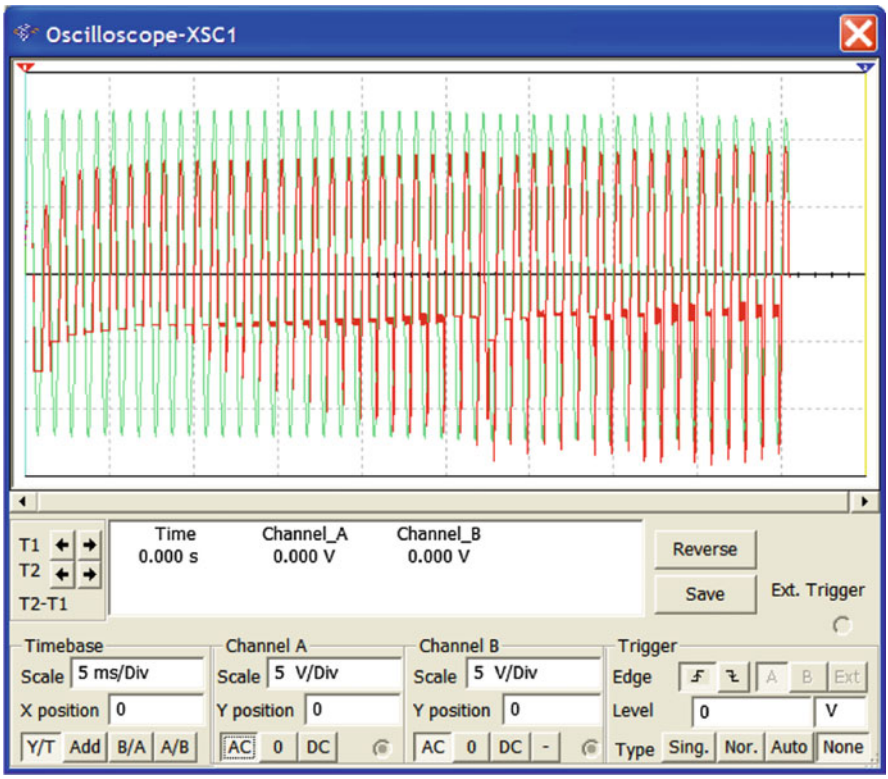

Fig. 4.4 Output chaotic signal $\upsilon_y = \upsilon_y(t)$ (*red – dark grey line*) across R_1 and $\upsilon_x = \upsilon_x(t)$ across R_2 for the RLT circuit of Fig. 4.1b. For $f_1 = 1$ KHz and $f_2 = 900$ Hz, $\upsilon_x = \upsilon_x(t)$ is periodic although $\upsilon_y = \upsilon_y(t)$ is chaotic

information in a signal time series appears symmetrical to the last information available before the observation on t + 1.

According to [11–13], the first step is to define a starting training period and divide that period on different vectors $\upsilon_{y_T}^m$ of size m, where t = m,. . .,T. The value of T is the number of observation data on the training period. The term m is also defined as the embedding dimension of the time series. For notation purposes, the last vector available prior the observation be forecasted will be called $\upsilon_{y_T}^m$ and will be the emitter voltage, while the other pieces will be addressed as $\upsilon_{y_t}^m$. The second step is to select the k most similar pieces to $\upsilon_{y_T}^m$. In a formal notation for the correlation method this means that we search for the k pieces with the highest value of

$$|\rho|(\upsilon_{y_i}^m, \upsilon_{y_t}^m) + |\rho|(\upsilon_{x_i}^m, \upsilon_{x_t}^m) \tag{4.1}$$

i.e., maximizing (4.1), where the term $|\rho|(\upsilon_{y_i}^m, \upsilon_{y_t}^m)$ is the absolute correlation between the pieces $(\upsilon_{y_i}^m)$ and the piece located before the value to be forecasted $\upsilon_{y_T}^m$.

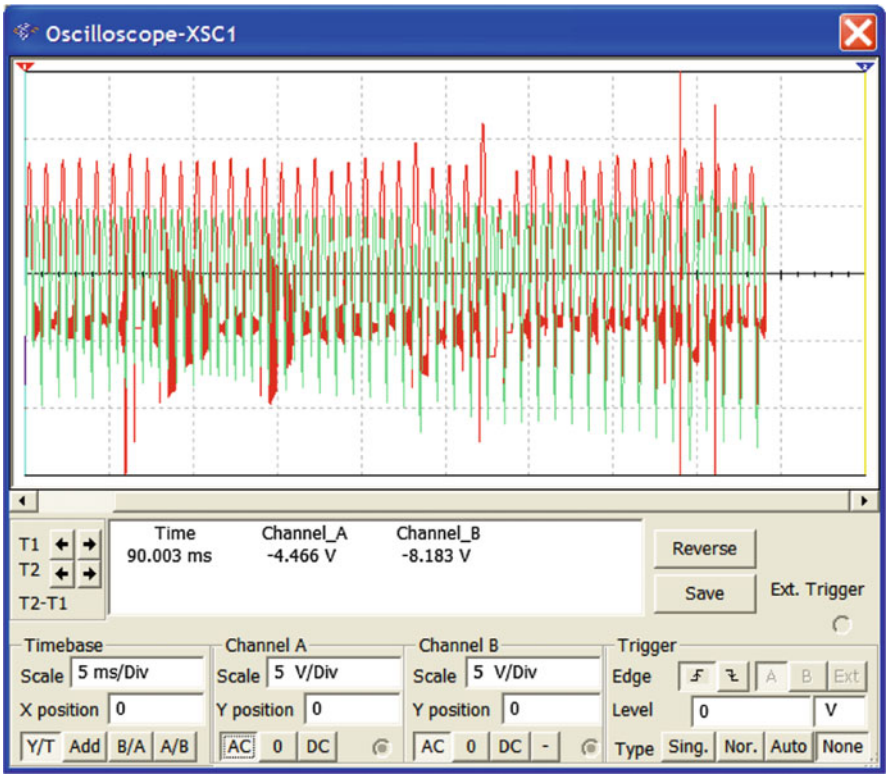

Fig. 4.5 Output chaotic signal $\upsilon_y = \upsilon_y(t)$ (*red –dark grey line*) across emitter resistor R_1 and $\upsilon_x = \upsilon_x(t)$ across collector resistor R_2 for the RLT circuit of Fig. 4.1b. For $f_1 = 1$ KHz and $f_2 = 995$ Hz both are chaotic

Following the above notation, $\upsilon_{x_t}^m$ are the historical pieces of the independent voltage time series at collector υ_{xt} and υ_{xT}^m is the piece located before the observation to be forecasted on υ_{yt}. The rational of using (4.1) is that the independent and the dependent series present regularity on the nearest neighbor location. Thus, using υ_{xt} to model υ_{yt} leads the algorithm to find the location of the k nearest neighbor of υ_{yT}^m [14, 15].

With the k pieces in hand, each of one of m observations, is necessary to understand in which way the k vectors can be used to construct the forecast on t + 1. There are several ways to be employed here including the use of an average or a tricube function [6]. According to [6] we calculate the following summation

$$\hat{\upsilon}_{y_{T+1}} = \hat{a}_0 + \hat{a}_1 \upsilon_{y_{T-1}} + \hat{a}_2 \upsilon_{y_{T-2}} + \ldots \hat{a}_m \upsilon_{y_{T-m}} \tag{4.2}$$

for which the coefficients $\hat{a}_0, \hat{a}_1, \ldots . \hat{a}_m$ are those derived from the estimation of a linear model with the dependent variable $\upsilon_{y_{i_r+1}}$ and the explanatory variables

$\upsilon_{y_{i_r}}^{m} = (\upsilon_{y_{i_r}} \upsilon_{y_{i_r-1}} \ldots \upsilon_{y_{i_r-m+1}})$, with r varying from 1 to k. In order to facilitate the understanding of such regression, (4.2) can be presented in matrix form as:

$$\begin{bmatrix} \upsilon_{yi_1+1} \\ \upsilon_{yi_1+1} \\ . \\ . \\ . \\ \upsilon_{yi_1+1} \end{bmatrix} = \hat{a}_0 + \hat{a}_1 \begin{bmatrix} \upsilon_{yi_1} \\ \upsilon_{yi_2} \\ . \\ . \\ . \\ \upsilon_{yi_k} \end{bmatrix} + \hat{a}_2 \begin{bmatrix} \upsilon_{yi_1-1} \\ \upsilon_{yi_2-1} \\ . \\ . \\ . \\ \upsilon_{yi_k-1} \end{bmatrix} + \ldots + \hat{a}_{m-1} \begin{bmatrix} \upsilon_{yi_1-m+1} \\ \upsilon_{yi_2-m+1} \\ . \\ . \\ . \\ \upsilon_{yi_k-m+1} \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ . \\ . \\ . \\ \varepsilon_3 \end{bmatrix}. \tag{4.3}$$

To further clarify (4.3), it is necessary to comprehend that the NN algorithm is non temporal. The values of $\upsilon_{y_{i_k+1}}$ are the voltage observations one period ahead of the pieces chosen by the correlation criteria defined earlier. The term $\upsilon_{y_{i_k-m}}$ indicates the first values of the k chosen pieces, while the term $\upsilon_{y_{i_k}}$ represents the last values of each chosen piece. Then, it is easy to see that the number of explanatory series in (4.3) will be m and each of them will have k observations. Term $\hat{a}_1$ in (4.3) is the coefficient aggregated to the last observation of the chosen series and $\hat{a}_2$ is the coefficient for all the second last observations of all k series. This logic for the coefficients continues until it reaches the first observation of all k chosen series, $\hat{a}_{m-1}$ and the values of the coefficients of (4.3) are then estimated by the minimizing the sum of the quadratic error $\sum_{i=1}^{k} \varepsilon_{\kappa}^{2}$. This procedure is executed in a loop manner until the point for which all forecasts on t + 1 are created.

4.2.2 Voltage Time Series Prediction

We now predict the emitter time series $\upsilon_y = \upsilon_y(t)$ in an out of sample manner for n = 15 time steps ahead. For this, we aim to find the nearest neighbors of collector voltage υy using the emitter voltage υx. Based on our previous work, [4], we consider the embedding dimension of $\upsilon_y = \upsilon_y(t)$ time series to be equal to m = 3. Then, if we apply the same procedure as in [4, 16–18], we will find that the embedding dimension of $\upsilon x = \upsilon x(t)$ time series is also m = 3. Hence we can apply the above procedure into our time series. In the time series shown in Fig. 4.3, we use T = 1,502 data points, i.e. 1,502 time indexes with sample rate $\Delta t = 1.9 \times 10^{-8}$ s. Moreover, by keeping in mind that m = 3, the size of vectors will be $\upsilon_{y_{1502}}^{3}$ and $\upsilon_{x_{1502}}^{3}$, [6, 19]. For the number of nearest neighbors to be used in the construction of the forecasts we found that k = 17 results to the best outcomes. We start the forecasting from half of time series, i.e. from 750 time index. The training period of the algorithm includes the values from 1 to 750 and this will be the training period for the forecasted value of 751. For the forecast of 752, the training period will be 1–752, meaning that each time a new observation is available the algorithm will add it to the training period. The out samples forecasts

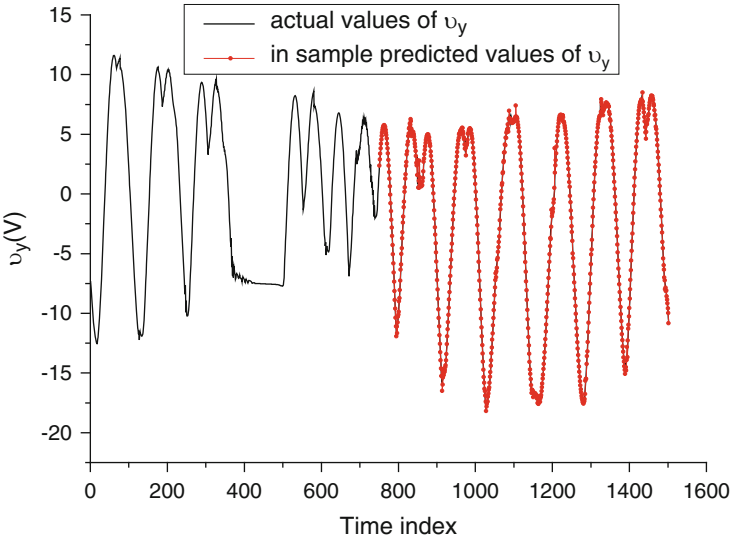

Fig. 4.6 Actual (*black line*) and predicted (*red–light grey line*) in sample values of output chaotic signal $\upsilon_y = \upsilon_y(t)$ across the emitter

are build with the whole modeled series υ_y. Actual and predicted time series for in sample prediction are presented in Fig. 4.6, where the black [dark] line represents the actual values and the red [light grey] line represents the predicted values. The actual and the out of sample predicted time series for 15 time steps ahead are presented in Fig. 4.7 with the same notation.

4.3 An Externally Triggered Single Transistor Chaotic Circuit Operating in Its Reverse Active Region

The complete layout of the considered circuit is shown in Fig. 4.8a. It consists of a basic common emitter configuration of a BC107BP npn-type BJT transistor with an emitter degeneration resistor $R_2 = 33\ \Omega$ and a collector resistor $R_1 = 3.3\ k\Omega$ in series. The circuit is driven by an input sinusoidal voltage with amplitude V_0 as applied through an inductor $L = 55\ \mu H$ directly to the transistor base and is powered supplied by a DC voltage $V_1 = 12$ Volts connected to the transistor emitter through R_2 rather than to the collector through R_1, which is the common practice, since our intension is to force the BJT to operate in its reverse active region. Under these conditions, we examine its operation by monitoring the collector voltage VC(t) waveform as it appears across the grounded resistor R_1.

In particular, the considered output voltage VC(t), i.e. the voltage across the collector resistor R_1, depends on the collector current iC(t) which is an important circuit parameter, since it will become chaotic when or if the circuit exhibits chaotic operation. This can be seen in the upper waveform of Fig. 4.9, which depicts the

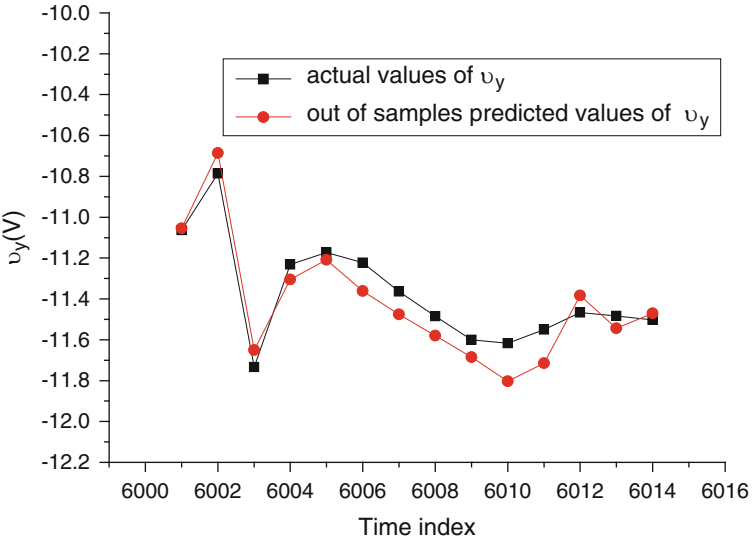

Fig. 4.7 Actual (*black line*) and predicted (*red–light grey line*) out of sample values of output chaotic signal $\upsilon_y = \upsilon_y(t)$ across emitter

obtained chaotic time series of the output voltage $VC(t) = iC(t)R_1$ when the amplitude V_o of the externally applied input sinusoidal voltage takes the value $V_o = 13$ Volts and frequency $f_1 = 3$ KHz. Of particular interest, however, is the lower waveform shown in Fig. 4.9, which is the input signal as it appears at the circuit input, i.e. prior the inductor L. As it can be seen, the input signal negative semi-period is heavily distorted due to the applied emitter DC voltage V_1 and the reverse biasing the base—emitter junction although the base-collector junction remains forward biased. This mode of BJT operation is generally known as operation in the reverse active region and, thus, the distorted part of the input signal corresponds to a "reverse saturation state" for that mode of operation. Hence, the applied input signal as well as the reversely active operation of the transistor in the considered RLT circuit result to the depicted output chaotic signal VC(t). And it is for this exact reason that the collector voltage is not inverted with respect to the input signal as it would have been expected to be for a transistor operating in its nominal mode of operation, thoroughly studied in other similar circuits in the past [20, 21].

Following the above and aiming to explore the ability to externally control the possible chaotic operation of the considered RLT circuit, we assume that the expected base region width-modulation [known also as "Early effect"] when a BJT operates in the reverse active region may affect more or less heavily the considered circuit's chaotic operation. This mode of BJT operation compared to the nominal forward active region refers to a quite different physical insight in the generation of the depletion region of the reverse biased base-emitter junction and its width-modulation within the base region during the external sinusoidal triggering through the forward biased base-collector junction. In this respect, we

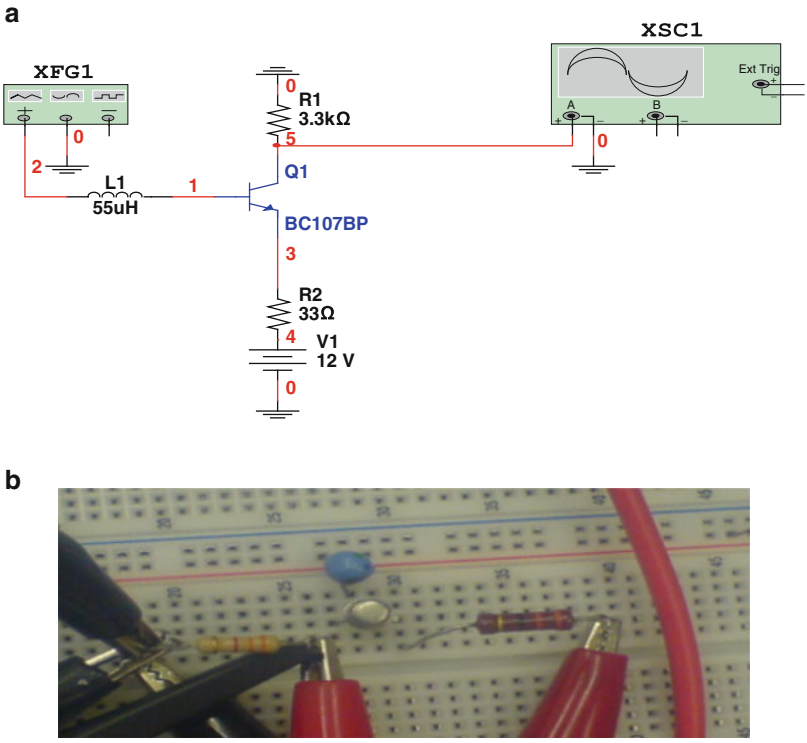

Fig. 4.8 (**a**) Schematic of the considered RLT circuit, (**b**) Laboratory implementation

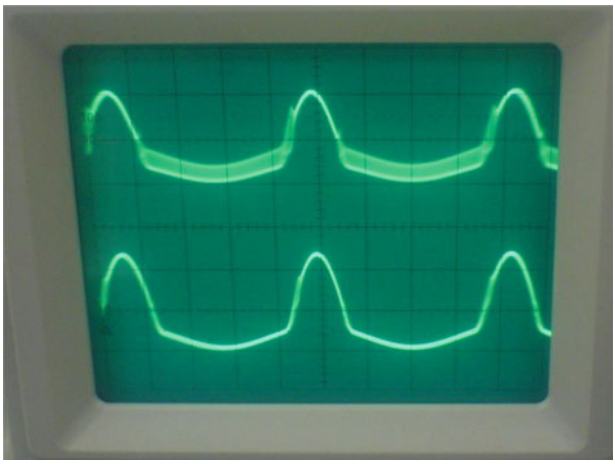

Fig. 4.9 Output chaotic signal VC(t) across R_1 (*upper waveform*) and input voltage (*lower waveform*)

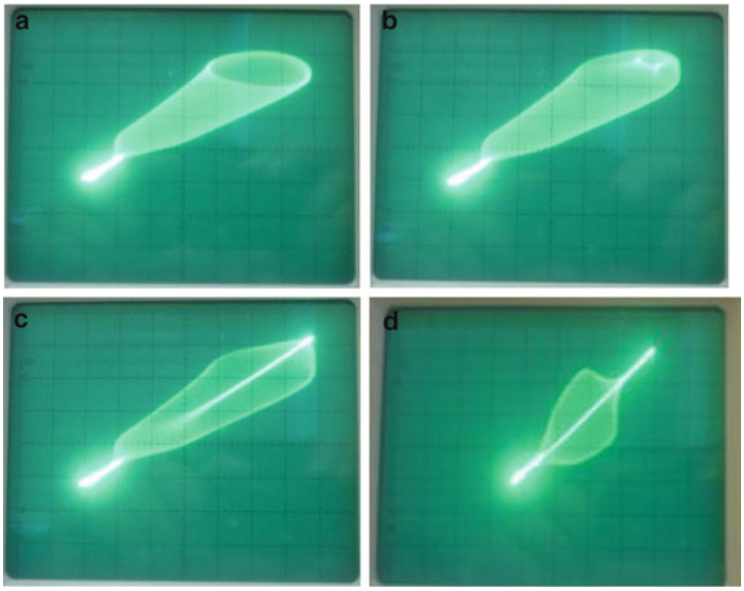

Fig. 4.10 The obtained phase portrait of the output signal VC(t) across R_1 with respect to the amplitude V_0 of the input signal: (**a**) periodic attractor with $V_0 = 10$ Volts, (**b**) semi periodic attractor with $V_0 = 11.5$ Volts, (**c**) chaotic attractor with $V_0 = 12.5$ Volts, and (**d**) chaotic attractor with $V_0 = 13$ Volts

experimentally investigated whether or not the achieved chaotic operation of our circuit can be controlled by means of the applied input signal amplitude V_o. This is shown in Fig. 4.10, in which the phase portraits of the output signal VC(t) are depicted with respect to the various values of the input signal amplitude V_o, while in Fig. 4.11a, b we show the output chaotic time series of the finally achieved chaotic output signal as obtained in real time on both analogue and digital oscilloscopes.

4.3.1 Reconstruction of the Phase Space

For a scalar time series as it is in our case the recorded voltage time series across the collector resistor, VC(t) = Vt where t = 1,2,3,..N, the phase space can be reconstructed using the method of delays [18, 22–26]. The basic idea of that method is that the evolution of any single variable of a system is determined by the other variables that it interacts with. Information about the relevant variables is thus implicitly contained in the history of any single variable.

Based on this, an "equivalent" phase space can be reconstructed by assigning an element of the time series V_t and its successive delays as coordinates of a new vector time series

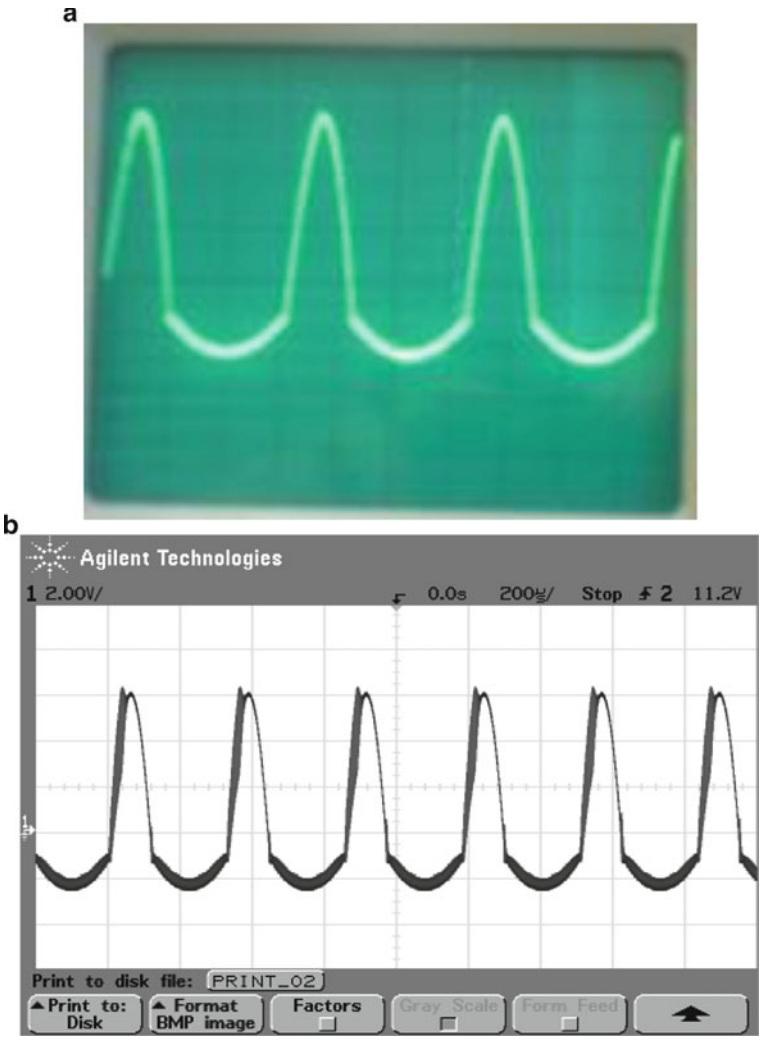

Fig. 4.11 Output chaotic signal VC(t) across R_1 as shown in real time on (**a**) an analogue and (**b**) a digital oscilloscope

$$\vec{X}_l = \{V_t, V_{t-\tau}, V_{t-2\tau}, ..., V_{t-(m-1)_\tau}\} \tag{4.4}$$

where τ is referred to as the delay time and for a digitized time series is a multiple of the sampling interval used, while m is termed as the embedding dimension. Figure 4.12 depicts the recorded voltage time series data across the collector resistor R_1 with sample rate $\Delta t = 0.5$ μs, already shown in real time in Fig. 4.11b.

The dimension m of the reconstructed phase space is considered as sufficient for recovering the object without distorting any of its topological properties, though it

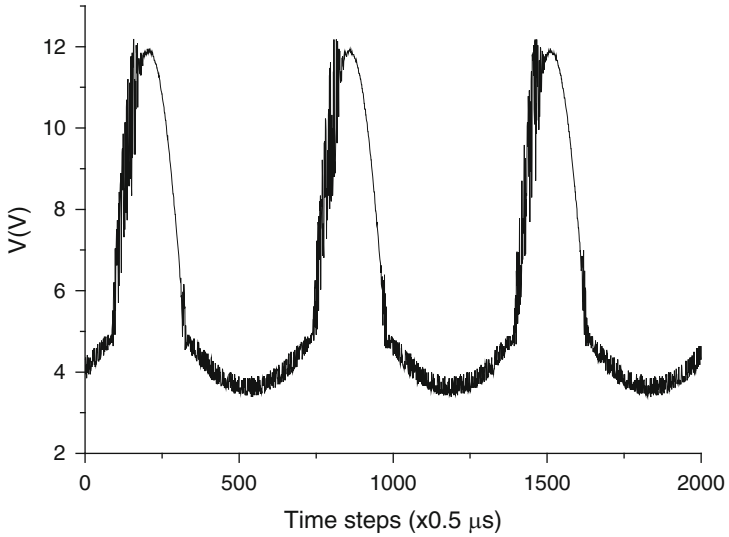

Fig. 4.12 The recorded data of the output chaotic signal V = VC(t) for the RLT circuit of Fig. 4.8

may be different from the true dimension of the space where this object lies. Here, the number of the experimental points is N = 2,000, while considering the m dimensional space, each vector $\vec{X}_l$ represents a point of the m dimensional phase space in which the attractor is embedded each time. In (4.1), τ is the time delay determined by the first minimum of the mutual information function I(τ), [25], and is defined as $\tau = \ell \Delta t$ with $\ell = 1,2,\ldots,N$ and Δt being the sample rate. As shown in Fig. 4.13, in our case the mutual information function I(τ) exhibits a local minimum at τ = 4 time steps and, thus, we shall consider τ = 4 to be the optimum delay time.

We then need to determine the Theiler window, [18]. As Theiler pointed out, if the temporally correlated points are not neglected, then spuriously low dimension estimate may be obtained [17, 18, 27]. However, since there is no concrete rule of how to choose W, we may choose to take the time at which the autocorrelation function (ACF) has dropped to 1/e of its initial value [18, 28]. As shown in Fig. 4.14, the correlation length τc is equal to τc = 96 and, thus, W = τc = 96 time lags.

Embedding theorems show that there is always a dimension m for which the geometric object formed by X_t is equivalent to the original trajectory V_t. A technique to estimate the optimal delay dimension m is by looking for false neighbors in phase space [9, 29]. More specifically, the method is based on a fact that when embedding dimension is too low, the trajectory in the phase space will cross itself. If we are able to detect these crossings, we may decide whether the used m is large enough for correct reconstruction of the original phase space [i.e. when no intersections occur] or not. When intersections are present for a given m, the embedding dimension is too low and we have to increase it at least. Then, we test

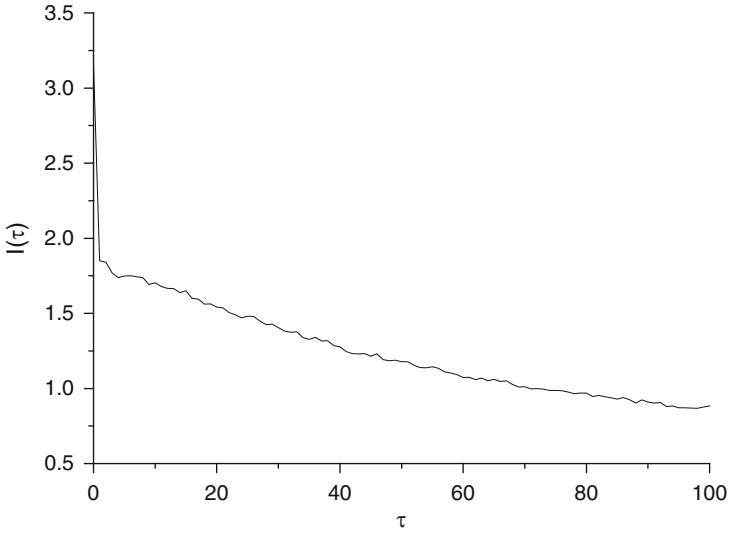

Fig. 4.13 Average mutual information I(τ) versus time delay τ

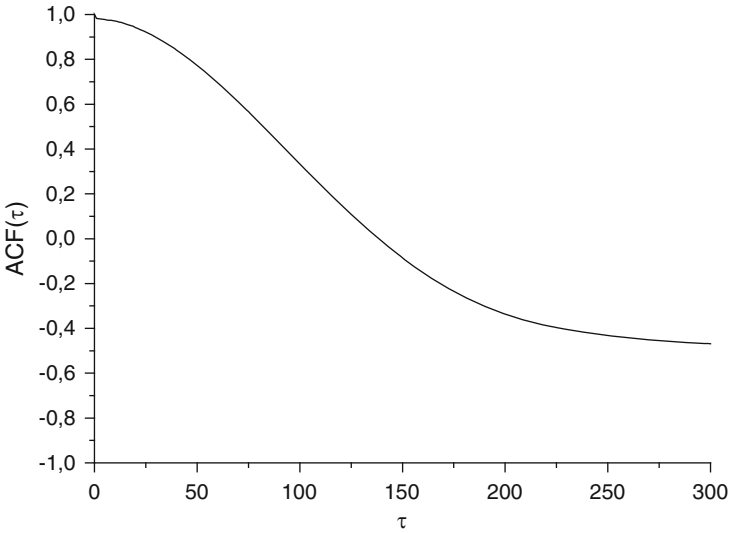

Fig. 4.14 Autocorrelation function ACF(τ) versus time delay τ

the eventual presence of self-crossings again until find a characteristic embedding dimension at which the trajectory in the phase space will not cross itself. This dimension is denoted by m_0.

A noise free time series, theoretically, would not have any false neighbors for an embedding dimension m larger than a m_0. This embedding dimension m_0 is as

we said before the embedding dimension at which the trajectory in the phase space will not cross itself. If we increase more the embedding dimension and repeat the procedure, we would not have any false neighbors. Conversely, for real-life data, the false neighbors for the optimum embedding dimension can be considered as "noisy" samples (with respect to the assumed deterministic relationship). These noisy samples can be removed by removing the false neighbors at dimension m_0 and the phase space can be said to be de-noised with respect to the deterministic relationship assumed in the phase space reconstruction of the system. We implement the false nearest-neighbor algorithm before and after applying the de-noising technique, described in [30–32], with delay time $\tau = 4$, number of neighbors equal to 20, embedding dimension equal to five, and calculating the distance between two points, using the Euclidian norm, in order to judge whether these points are false neighbors at upper dimensions or not. The noise will then be reduced by 46.1%. We have chosen the nonlinear noise-reduction method proposed by H. Kantz, see [30–32], instead a wavelet de-noising procedure since it is easily applied and not difficult to choose the parameters of this algorithm, even though we use no other information about the underlying dynamics than the data themselves. The noise reduction is very robust with respect to changes in the choice of parameter.

The practical realization of the described method is based on testing of the neighboring points in m-dimensional phase space. Typically, we take a certain amount of points in the phase space and find the nearest neighbor to each of them. Then we compute distances for all these pairs and also their distances in (m + 1)-dimensional phase space. The rate of these distances is given by

$$P = \frac{\left\| \vec{X}_i(m+1) - \vec{X}_{n(i)}(m+1) \right\|}{\left\| \vec{X}_i(m) - \vec{X}_{n(i)}(m) \right\|} \tag{4.5}$$

where $\vec{X}_i(m)$ represents the reconstructed vector as described in (4.1), belonging to the i-th point in the m-dimensional phase space and index n(i) denotes the nearest neighbor to the i-th point. If P is greater than some value P_{max}, we call this pair of points false nearest neighbors [i.e. neighbors, which arise from trajectory self-intersection and not from the closeness in the original phase space]. In the ideal case, the value of m is found when the number of false neighbors falls to zero. For this purpose, we compute the rate of false nearest neighbors in the reconstructed phase space using the formula

$$\left| V_{i+m\tau} - V_{n(i)+m\tau} \right| \geq R_A \tag{4.6}$$

where RA is the radius of the attractor,

$$R_A = \frac{1}{N} \sum_{i=1}^{N} |V_i - \bar{V}| \tag{4.7}$$

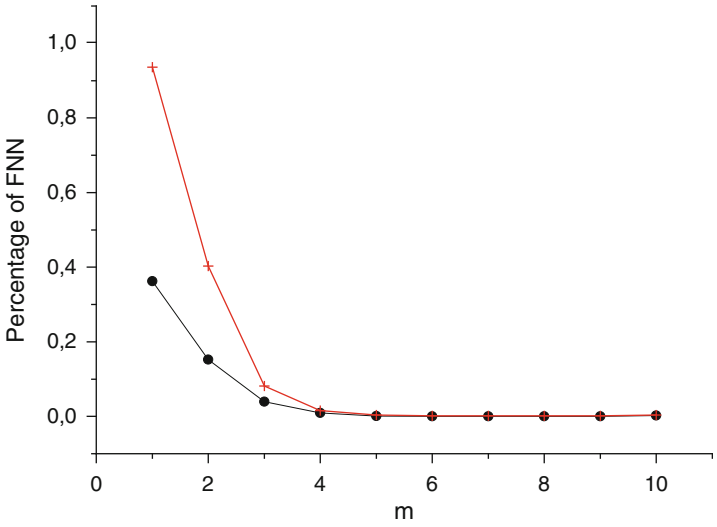

Fig. 4.15 Percentage of false nearest neighbors number FNN versus m before (*black dot line*) and after (*red line with crosses*) noise reduction

and

$$\bar{V} = \frac{1}{N} \sum_{i=1}^{N} Vi \tag{4.8}$$

is the average value of time series.

When the following criterion

$$P \geq P_{\max} \tag{4.9}$$

is satisfied, then it can be used to distinguish between true and false neighbors. The dimension m will be found when the false nearest neighbors percentage falls below some limit, typically set to 1% [33], and, thus, by choosing $P_{max} = 10$ and using Matlab code we finally calculate the quantity P. The so obtained results are shown in Fig. 4.15 indicating that the application of the FNN method yields an embedding dimension m_0 equal to five.

4.3.2 Determination of the Strength of Chaos

In order to determine the strength of the observed chaotic behavior, we calculate the largest Lyapunov exponent, which measures the divergence of nearby trajectories. As the system evolves, the sum of a series of attractor point values [in each dimension] will either converge or diverge. Lyapunov exponents measure this

convergence/divergence rate in each dimension, and a chaotic system will exhibit trajectory divergence in at least one dimension. Thus, a positive Lyapunov exponent is a strong indicator of chaos [34–36]. The method used in calculating the largest Lyapunov exponent is based on averaging the local divergence rates or the local Lyapunov exponents. Rosenstein et al. in [13] proposed a new method to calculate the largest Lyapunov exponent from an observed times series. Hence, after reconstructing the phase space using suitable values for τ and m, we can compute the logarithm of the average distance of a point X_{n0} in phase space with respect to all points X_n in its r-neighborhood. This is repeated for N number of points along the orbit so as to calculate an average quantity S known as the stretching factor and given by

$$S = \frac{1}{N} \sum_{n0=1}^{N} \ln \left[\frac{1}{|u_{X_{n0}}|} \sum |X_{n0} - X_n| \right] \tag{4.10}$$

where $|u_{X_{n0}}|$ is the number of neighbors found around point X_{n0}. In the case of chaotic dynamics, a plot of the stretching factor S against the number of points N will yield a curve with a linear increase at the beginning, followed by an almost flat region. The slope of the linear portion of the first part of this curve gives an estimate of the largest Lyapunov exponent. The method is accurate for small data sets because it takes advantage of all available data [28].

The value of r is taken as the data interval is divided by 200, and again, in order to avoid temporal correlations we use the Theiler window of 96. The magnitude lnS versus the number of points N shows the expected linear increase and then after a flat region. This is clearly shown in Fig. 4.16, where the slope value that corresponds to the largest Lyapunov exponent is obtained after the least-squares line fit and is found to be equal to $\lambda_{max} = 0.02048$ for $m = m_0 = 5$. Moreover, the dependence of λ_{max} on the embedding dimension m is shown in Fig. 4.17.

The predictability of a time series using phase space techniques can be considered as a means of testing the deterministic nature of a system. The prediction algorithm that we use here has already been presented in detail in a number of works [34–36]. This algorithm does a search for all points in phase space that are within a radius of ε from the point in question. This search space contains what are called "the example points" that are obtained from historical time series data. The delay vector for such a given time series is formed by using (4.1), and, then, given the measured voltage scalar sequence $V_1, V_2, \ldots V_N$, we can predict the k-step-ahead sample V_{N+k}.

The ε-neighborhood of the current point, X_N, is found with the radius ε being chosen based on Sauer's method [16, 27]. For an accurate prediction, we have to find all the nearest neighbors X_{i_k} of X_N, where k is between 1 and K. An average of these nearest neighbors is used to construct a predictor and to estimate the future value $\hat{V}_{i+N}$ as

$$\hat{V}_{i+N} = \frac{1}{K} \sum_{k=1}^{K} w_k V_{ik+1} \tag{4.11}$$

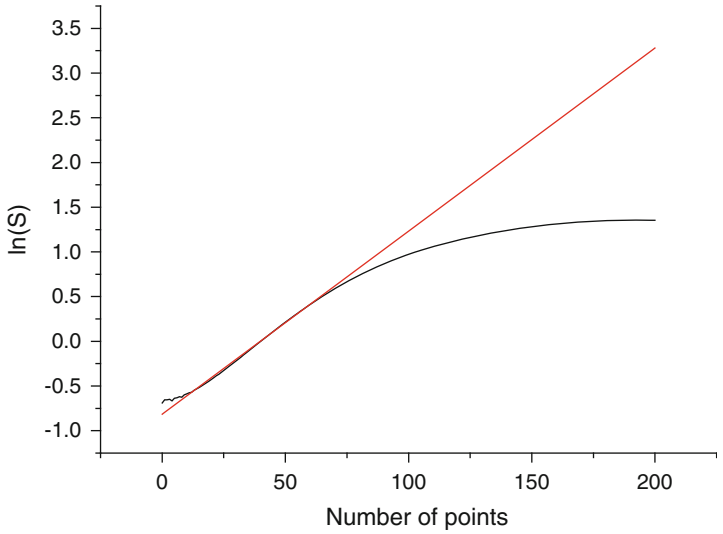

Fig. 4.16 Estimation of the largest Lyapunov exponent after the Rosenstein method. The portion of the curve used for the least square line fits the starting point up to the saturation point. The *straight line* in *red* represents the resulting least squares line fit with slope 0.02048 for $m = m_0 = 5$

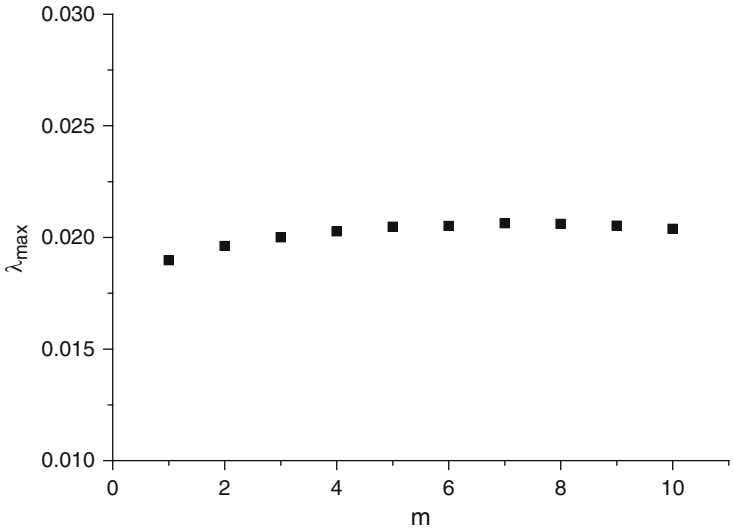

Fig. 4.17 The largest Lyapunov exponent λmax versus the embedding dimension m

with

$$w_k = \frac{e^{(-(d_k - d_{\min}))}}{\sum_{k=1}^{K} e^{(-(d_k - d_{\min}))}} \tag{4.12}$$

where d_k is the distance between X_N and X_{i_k}, d_{min} is the minimum value among [d_k with k = 1,2...K]. In this way, the predicted value is estimated by introducing a distance dependent weight and a weighted averaging algorithm. The predicted point $\hat{V}_{i+N}$ can be used for the reconstruction of vector $\hat{X}_{N+1}$ with the predicted values, then it is set as the new starting vector and the process is repeated. In fact, the [approximate] predictable horizon Δt_{max} is a function of the largest Lyapunov exponent [25, 27, 28], being given by

$$\Delta t_{\max} = \frac{1}{\lambda_{\max}} \tag{4.13}$$

Next, we consider how the optimum radius of the neighborhood, i.e., the parameter ε, is selected to make an accurate prediction. Increasing the value of ε would then increase the enclosed phase space points. According to [37] the number of the neighbors, K = 2–6 m_0, in the ε-neighborhood can lead to an improved prediction accuracy. We applied this procedure to the real noisy Voltage data which have been monitored with sample rate Δt = 0.5 μs [see Fig. 4.13]. The largest Lyapunov exponent, based on (4.13), suggests that the maximum window for accurate predictions is about 49 time steps, [38]. The parameters for time series prediction are τ = 4 and m = m_0 = 5. The obtained results are shown in Fig. 4.18. As it can be seen, we tried to predict 200 time steps ahead, but due to the Lyapunov exponent limit the prediction is much less accurate beyond the limit of 49 time steps.

The previous results are more clearly visualized in Fig. 4.19, where the Normalized Mean Squared Error (NMSE) is shown. The (NMSE) is calculated by

$$NMSE = MAX\left(\frac{\sum_{i=1}^{NP}\left(\tilde{V}_i - V_i\right)^2}{\sum_{i=1}^{NP}\left(\bar{V}_i - V_i\right)^2}, \frac{\sum_{i=1}^{NP}\left(\tilde{V}_i - V_i\right)^2}{\sum_{i=1}^{NP}\left(V_{i-1} - V_i\right)^2}\right) \tag{4.14}$$

where $\tilde{V}_i$ is the predicted value, Vi is the actual value, $\bar{V}$ is the average actual value, and NP is the range of values in the prediction interval. We can then see that the NMSE has been well increased after the predictable horizon Δt_{max} limit.

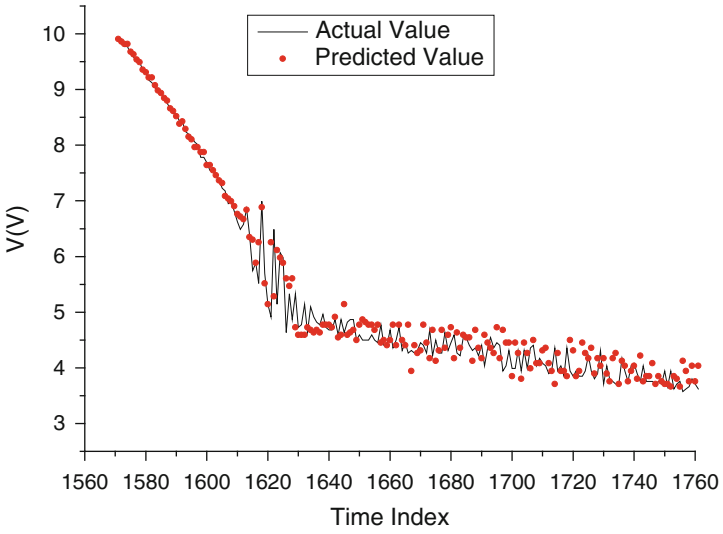

Fig. 4.18 Multi-step prediction of voltage oscillations across the collector resistor R_1 of the circuit of Fig. 4.8a

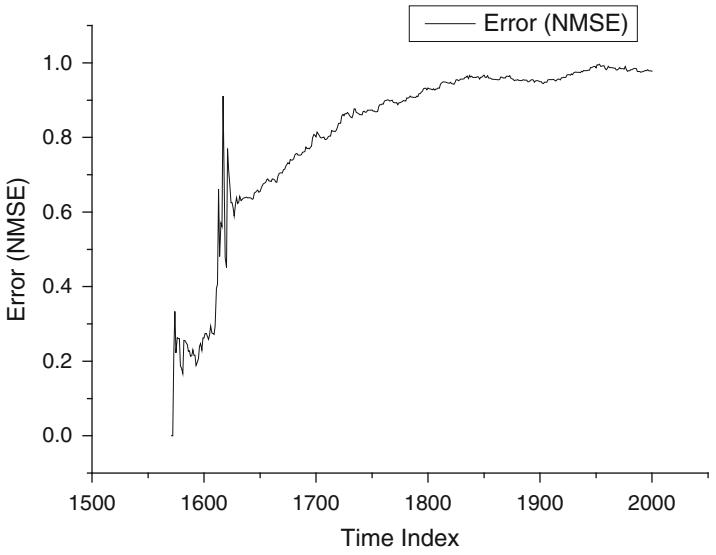

Fig. 4.19 Mean squared error of our predictor normalized by the mean squared error of the random walk predictor for multistep prediction

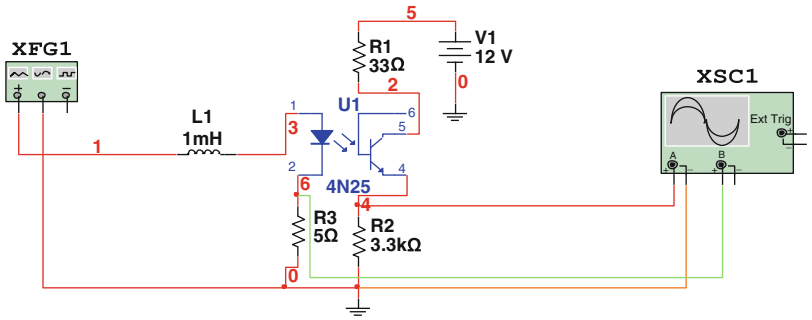

Fig. 4.20 Multisim simulation of the optocoupler circuit

4.4 Chaotic Optocoupler Circuit

The complete layout of the considered circuit is shown in Fig. 4.20. It consists of a 4 N25 optocoupler with basic common emitter configuration with an emitter degeneration resistor $R_2 = 3.3$ KΩ and a collector resistor $R_1 = 33$ Ω in series with the required DC power supply $V_1 = 12$ Volts. The circuit is driven by an input sinusoidal voltage with amplitude $V_0 = 13$ Volts as applied through an inductor $L = 1$ mH directly connected to LED. Clearly, considering the existing circuit resembles to a typical resistor-inductor-diode (RLT) circuit which is known as a simple chaotic circuit [1, 4, 39]. We examine its operation for chaos by means of the MultiSim circuit simulation environment, as illustrated in Fig. 4.20.

The simulated circuit operation is monitored by checking the voltages across emitter resistor R_2 and across LED resistor R_3. As it can be readily seen both voltages depend on the LED resistor R_3. Figure 4.21 shows the simulation obtained chaotic time series of the output signal across R_2 and across R_3 for input signal amplitude $V_1 = 13$ Volts and frequency f = 800 Hz.

Following the above, it must be noted that chaotic operation is obtained under various conditions and parameters' values since variation of R_3 may strengthen, weaken, or even destroy the achieved chaos. For example, if R_3 is increased, then chaos will weaken and vanish. For $R_3 = 500$ Ω there is no chaotic signal across R_2 and a weak chaotic signal across R_3 as shown in Fig. 4.22.

For $R_3 = 1$ KΩ both signals are not chaotic as shown in Fig. 4.23.

4.4.1 Non Linear Analysis

We now proceed to the analysis of the obtained chaotic time series. Using our data, with value $R_3 = 5$ Ω we construct a vector $\vec{X}_i$, I = 1..N, where N = 5,000 data values in the m dimensional phase space given in the form as in (4.4).

This vector, represents a point to the m dimensional phase space in which the attractor is embedded each time, where τ is the time delay $\tau = i\Delta t$ and $\Delta t = 0.1$

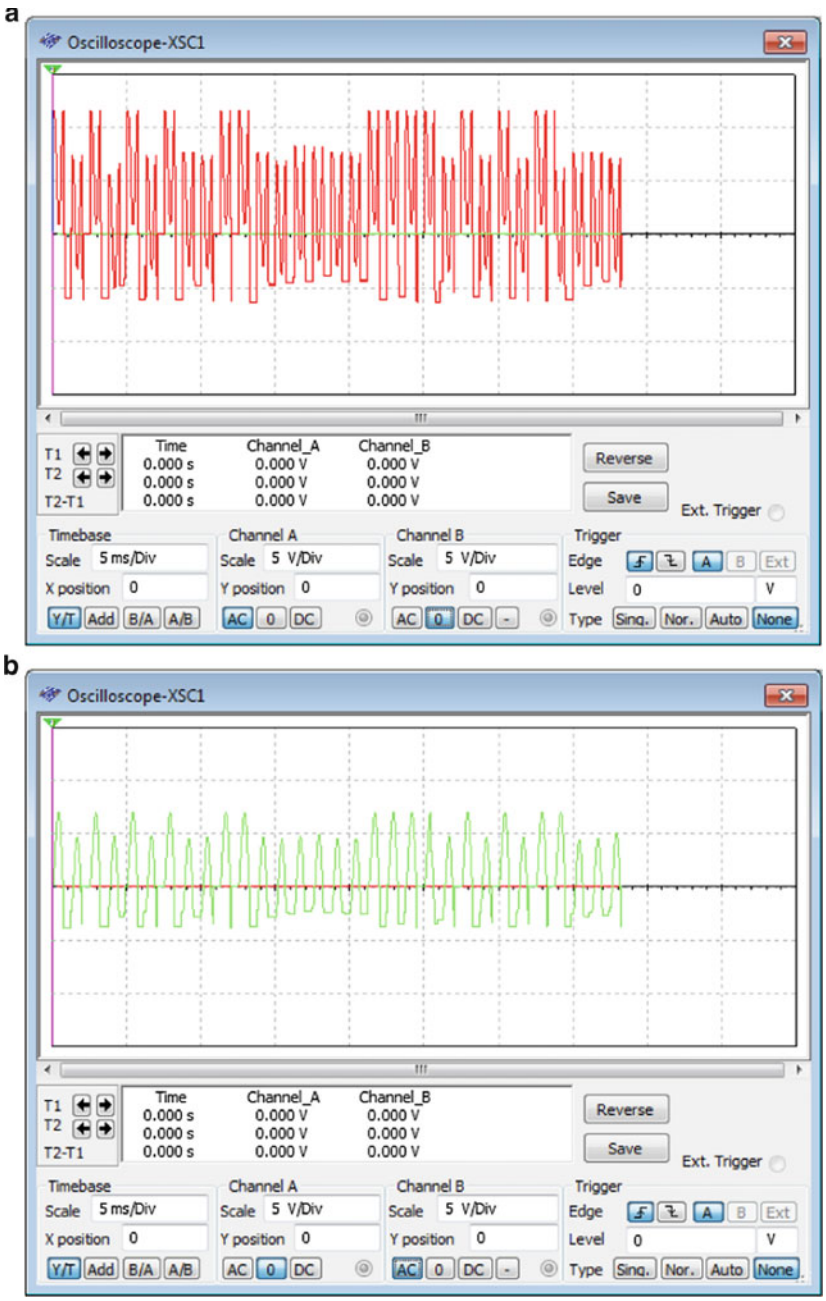

Fig. 4.21 Output chaotic signal V = V(t) across emitter resistor R_2 (**a**) and across LED resistor (**b**) for the optocoupler circuit of Fig. 4.20, for R = 5 Ω

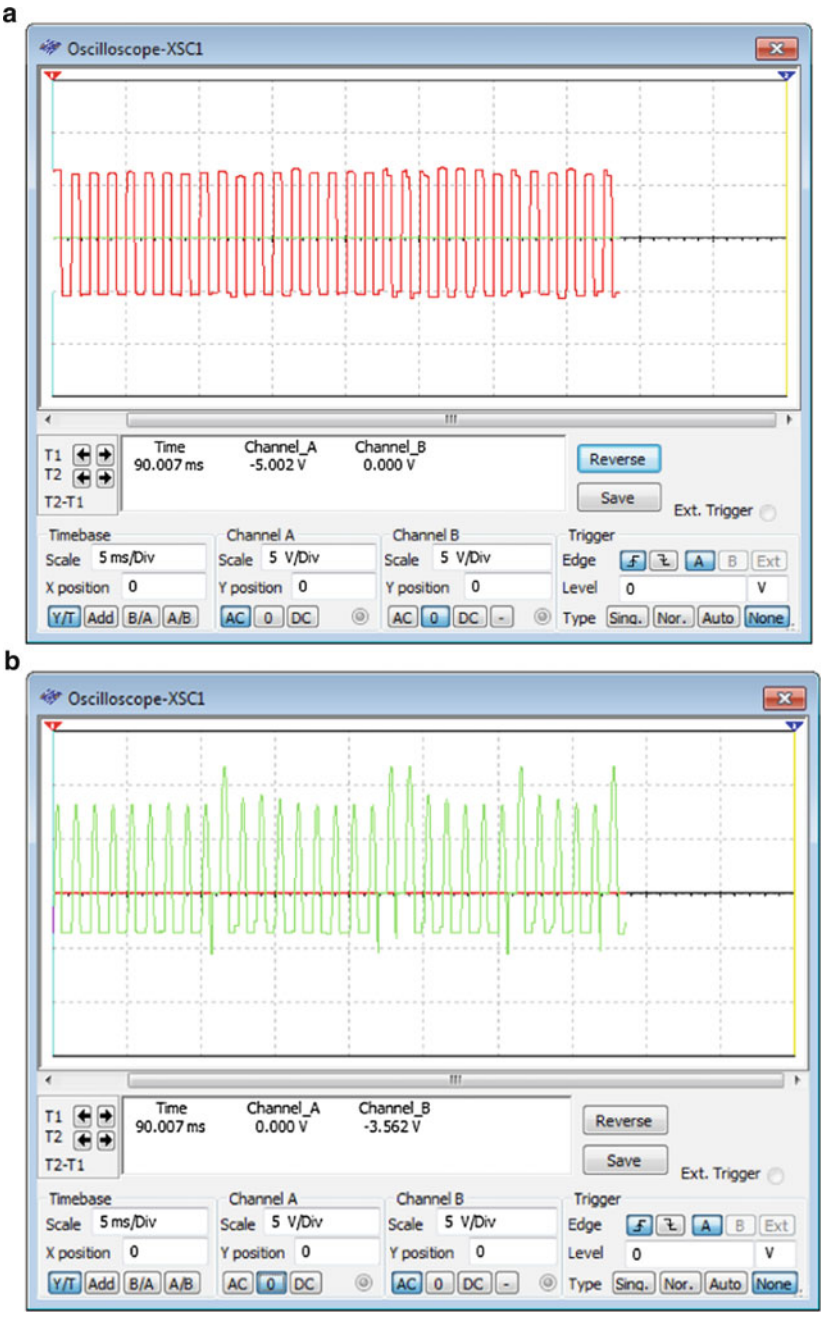

Fig. 4.22 Non chaotic signal V = V(t) across emitter resistor R_2 (**a**) and weak chaotic signal across LED resistor (**b**) for the optocoupler circuit of Fig. 4.1 for $R_3 = 500\ \Omega$

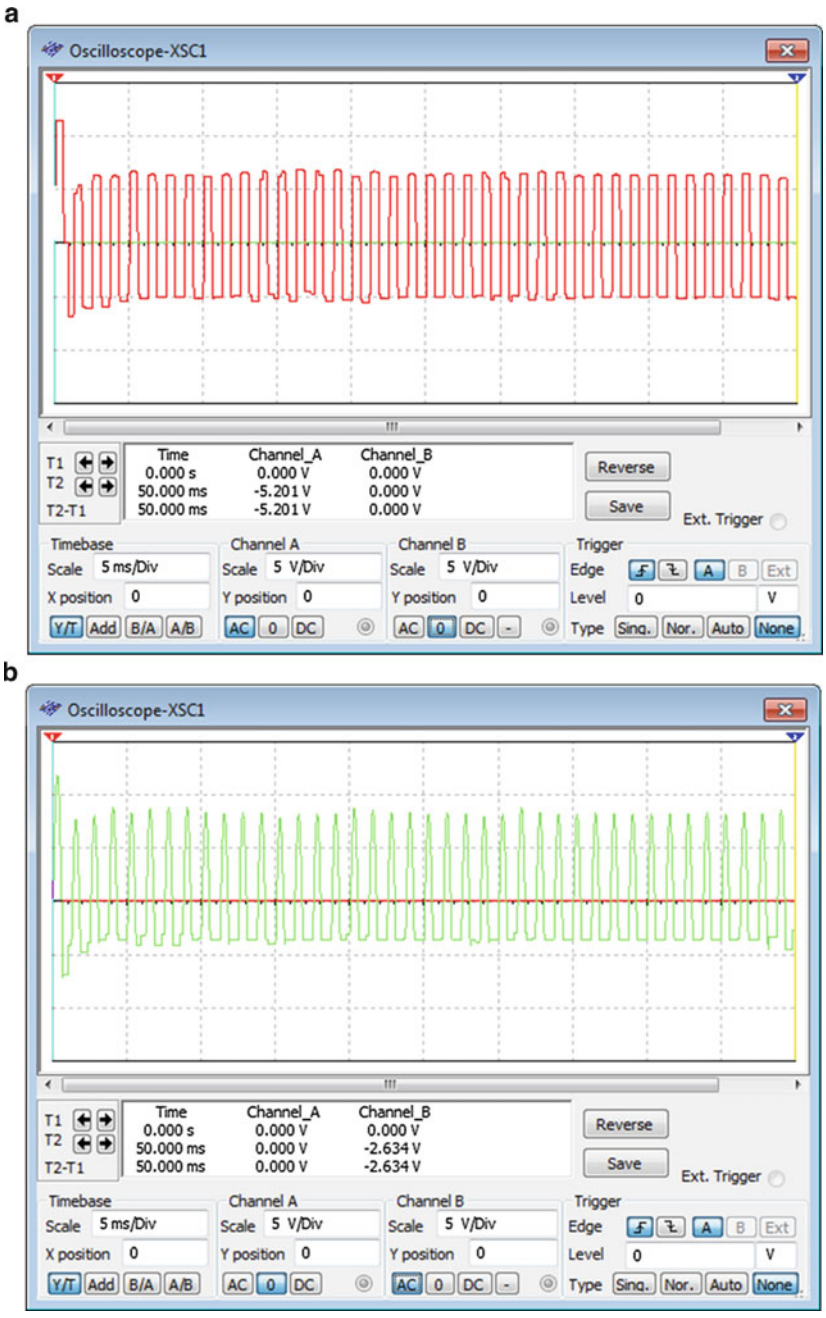

Fig. 4.23 Non chaotic signal V = V(t) across emitter resistor R_2 (*left panel*) and not chaotic signal across LED resistor (*right panel*) for the optocoupler circuit of Fig. 4.20 with $R_3 = 1$ KΩ

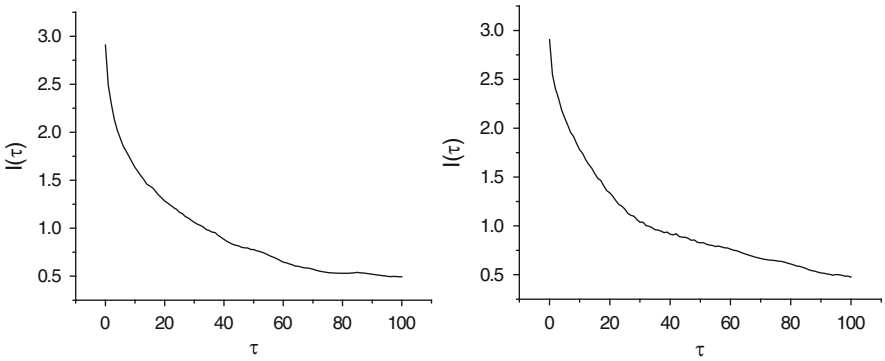

Fig. 4.24 Mutual information I versus the time delay τ. For chaotic signal across emitter resistor R_2 (*left panel*) and for chaotic signal across LED resistor (*right panel*) for the optocoupler circuit of Fig. 4.20 with $R_3 = 5\ \Omega$

μs is the sample rate. The element v_t represents a value of the examined scalar time series in time, in our case the voltage v across the resistor's R_2 and R_3 corresponding to the i-th component of the time series. Use of this method reduces phase space reconstruction to the problem of proper determining suitable values of m and τ. The choice of these values is not always simple, especially when we do not have any additional information about the original system and the only source of data is a simple sequence of scalar values, acquired from the original system. The dimension, where a time delay reconstruction of the phase space provides a necessary number of coordinates to unfold the dynamics from overflaps on itself caused by projection is called embedding dimension m. Using the average mutual information we can obtaining τ, less associated with linear point of view, and thus more suitable for dealing with nonlinear problems. The average mutual information, I(τ) [18] expresses the amount of information (in bits), which may be extracted from the value in time v_i about the value in time $v_{t+\tau}$. As τ, suitable for the phase space reconstruction, the position of the first minimum of I(τ) is usually used [18]. In this case $\tau = 77$ time steps for chaotic signal across emitter resistor R_2 and $\tau = 38$ steps for chaotic signal across LED resistor R_3 as shown in Fig. 4.24.

Next, we use the method of False Nearest Neighbors [18, 40] to estimate the minimum embedding dimension. The so obtained results are shown in Fig. 4.25, indicating that the application of the FNN method yields a minimum embedding equal to value 5 for both chaotic signals.

4.5 Chaotic Single Electron Transistor Circuit

Single-electron devices have received considerable attention recently, in particular, for possible device application in integrated circuits [41]. In general, a single-electron transistor (SET) is made of a small dot isolated from the source and

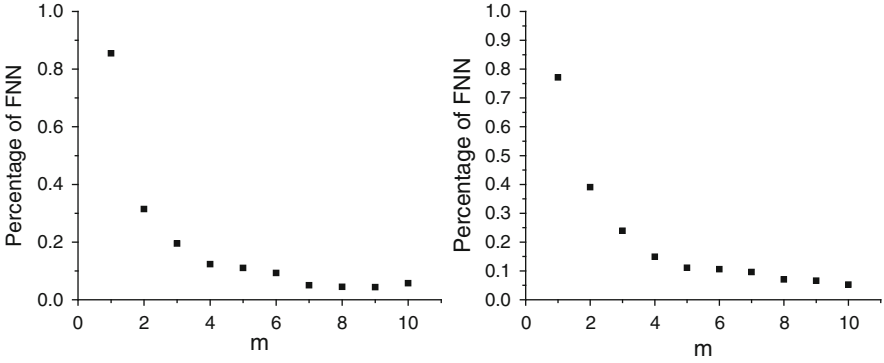

Fig. 4.25 False nearest neighbor ratio as a function of the embedding dimension. The false nearest neighbors become negligible after m = 5 for chaotic signal across emitter resistor R_2 (*left panel*) and for chaotic signal across LED resistor (*right panel*) for the optocoupler circuit of Fig. 4.20 with $R_3 = 5\ \Omega$

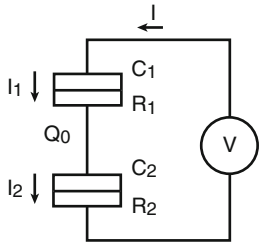

Fig. 4.26 Basic circuit of SET Q_0 is the background charge and C_1 and C_2 are the junction capacitances and R_1 and R_2 are the tunnel resistances

drain by two small tunnel junctions. Its basic circuit consists of two small area tunnel junctions connected in series and biased by a dc voltage source [42], Fig. 4.26.

Single electron transistor is considered as a circuit consisting of islands that are arbitrarily connected with tunnel junctions and capacitors as well as ideal voltage sources those control the circuit (Fig. 4.27). Single electron circuits are composed of small tunnel junctions, capacitances, and voltage sources. A tunneling electron can be described as a discrete charge due to stochastic nature of a tunneling event [43].

There are SPICE macro-modeling techniques for the compact simulation of single electron circuits [44] and also using a spice macro model single electron transistors can be used to design Chua type double scroll circuits [45]. We will concern to product chaotic voltage oscillations using a SET instead of a BC107BP npn-type BJT in a RTL circuit [4].

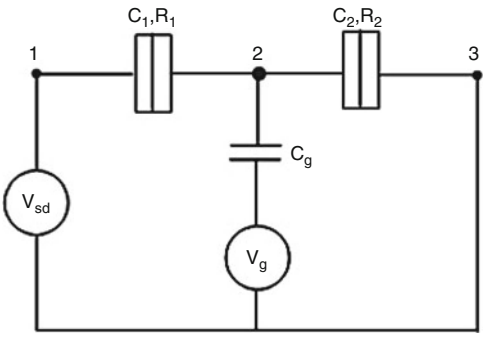

Fig. 4.27 Equivalent circuit of SET consisting of tunnel junctions with ideal voltage sources where C_g is the gate capacitance

4.5.1 Circuit Implementation

The complete layout of the considered circuit is shown in Fig. 4.1a. It consists of two 1 N3712 tunnel diodes agate capacitance $C_1 = 0.001$ μF a basic common emitter configuration with an emitter degeneration resistor $R_1 = 3$ KΩ and a collector resistor $R_2 = 30$ Ω in series. The circuit is driven by an input sinusoidal voltage with amplitude V_0 as applied through an inductor $L = 75$ μH directly to the transistor base. $V_3 = 1.5$ V and V1 = 3 mV are the biased dc voltage sources (Fig. 4.28).

Using an input sinusoidal voltage with amplitude $V_0 = 2.5$ Volts and frequency $f = 550$ Hz the voltage oscillations across the emitter resistor R_1 are shown in Fig. 4.29.

4.5.2 Estimation of Embedding Dimension

We now proceed to the analysis of the obtained chaotic time series. Using our data, we construct a vector $\vec{X}_i$, I = 1..N, where N = 5,000 data values in the m dimensional phase space given in the form as in (4.4).

This vector, represents a point to the m dimensional phase space in which the attractor is embedded each time, where τ is the time delay $\tau = i\Delta t$ and $\Delta t = 10$ μs is the sample rate. The element v_t represents a value of the examined scalar time series in time, in our case the voltage v across the resistor's R_1 corresponding to the i-th component of the time series. Use of this method reduces phase space reconstruction to the problem of proper determining suitable values of m and τ. The choice of these values is not always simple, especially when we do not have any additional information about the original system and the only source of data is a simple sequence of scalar values, acquired from the original system. The dimension, where a time delay reconstruction of the phase space provides a necessary number of coordinates to unfold the dynamics from overflaps on itself caused by

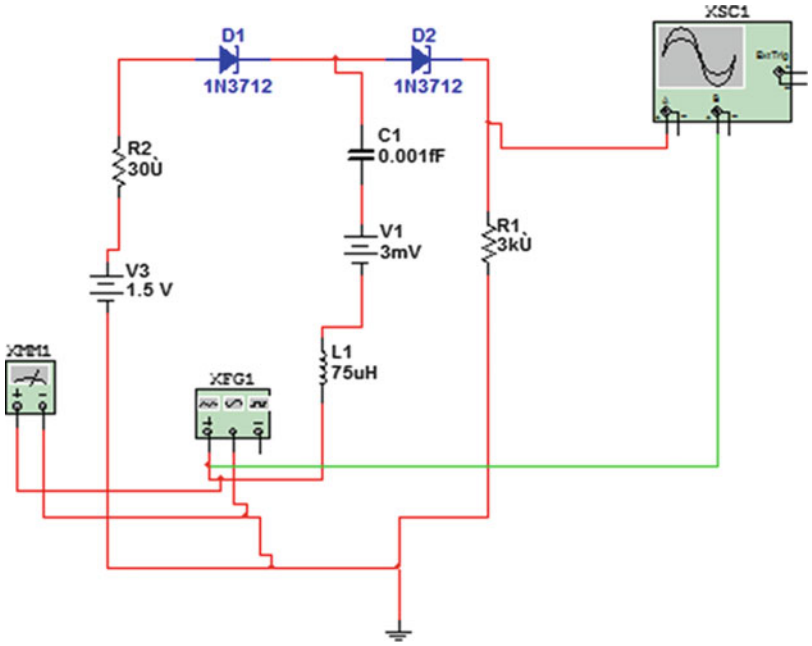

Fig. 4.28 Simulation environment of SET macro model

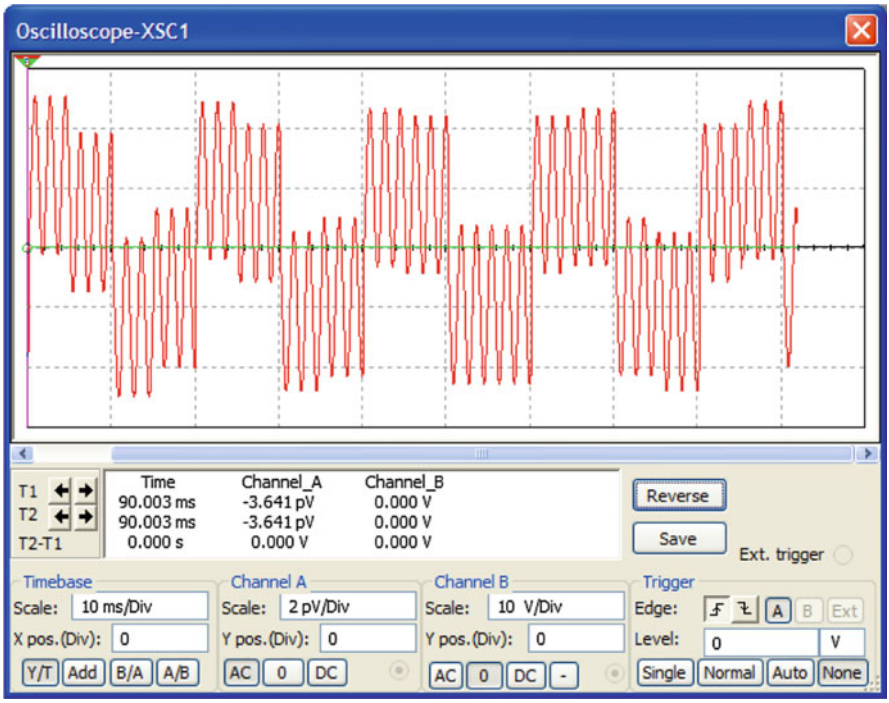

Fig. 4.29 Output chaotic signal across emitter resistor R_1 with f = 550 Hz. The control parameter is the frequency, because, by increasing f, the signal becomes periodic as shown in Fig. 4.30

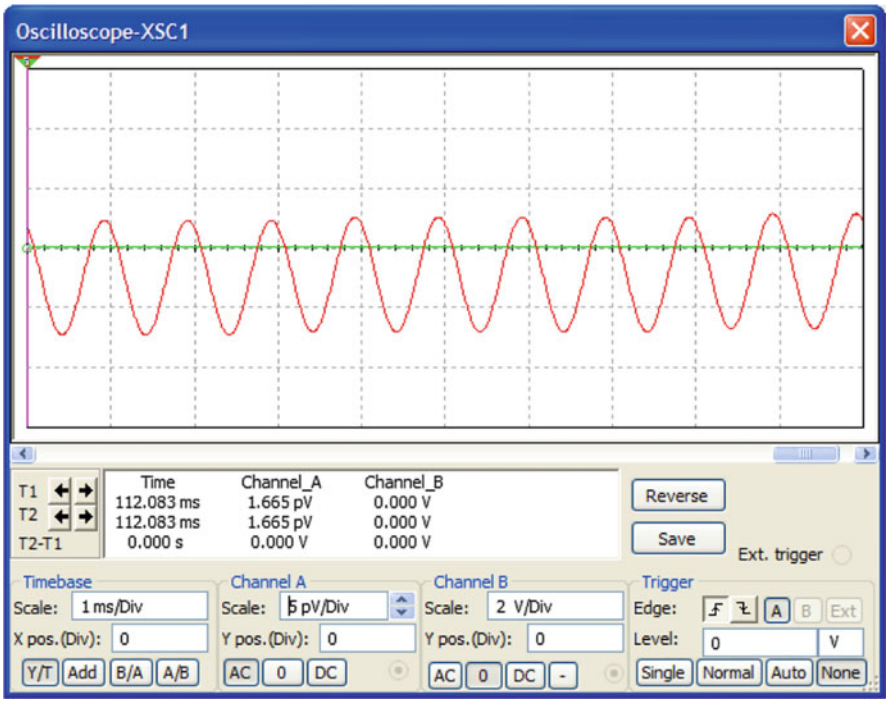

Fig. 4.30 Output periodic signal across emitter resistor R_1 with f = 1 KHz

projection is called embedding dimension m. Using the average mutual information we can obtaining τ, less associated with linear point of view, and thus more suitable for dealing with nonlinear problems. The average mutual information, I(τ) expresses the amount of information (in bits), which may be extracted from the value in time vi about the value in time $v_{t+\tau}$. As τ, suitable for the phase space reconstruction, the position of the first minimum of I(τ) is usually used. In this case τ = 14 time steps for chaotic signal across emitter resistor R_1 as shown in Fig. 4.31.

Next, we use the method of False Nearest Neighbors to estimate the minimum embedding dimension as we did in the Sect. 4.1. The so obtained results are shown in Fig. 4.32 indicating that the application of the FNN method yields a minimum embedding equal to value 3.

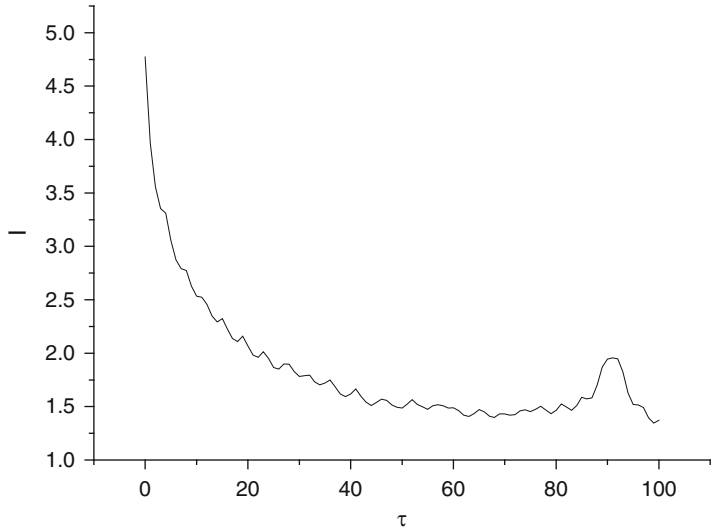

Fig. 4.31 Mutual information I versus the time delay τ

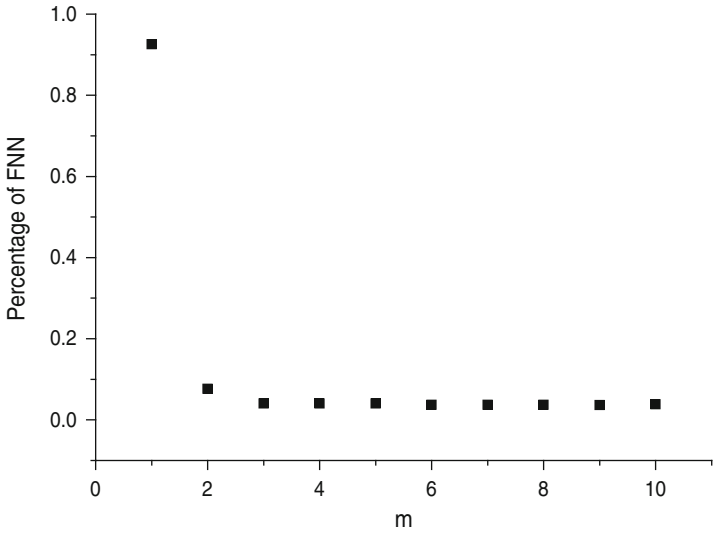

Fig. 4.32 False nearest neighbor ratio as a function of the embedding dimension. The false nearest neighbors become negligible after m = 5 for chaotic signal across emitter resistor R_1

4.6 Conclusions

It is clear that there are many circuits that produce chaotic oscillations based on transistor in various form a common BC107BP in a RTL circuit an optocoupler and even a single electron transistor. The use of an AC sinusoidal voltage supply in a RLT circuit strongly affects the chaotic or no state of both signals across emitter's and collector's resistors. The frequency of such a collector supply voltage is the control parameter. This can be used to dismiss or enhance the generated chaotic signals. In a microscopic manner the dynamic of carries at two p–n junctions which leads or not the chaotic state must be examined further based on the strong correlation between the voltages at emitter using the voltage at collector in predicting the first as affected by the later. The use of a simple RTL circuit operating in the reverse active region affects the chaotic state across the collector's resistor. We have verified and analyzed the experimentally obtained output voltage time series using the techniques based on phase space reconstruction. The results have shown that chaotic characteristics exist due to the positive largest Lyapunov exponent. The strange attractor folding in a 5 dimension phase space and the largest Lyapunov exponent dictate us that the maximum length of a prediction should be less than 49 time steps. The use of a simple RTL circuit operating in the reverse active region affects the chaotic state across the collector's resistor. We have verified and analyzed the experimentally obtained output voltage time series using the techniques based on phase space reconstruction. The results have shown that chaotic characteristics exist due to the positive largest Lyapunov exponent. The strange attractor folding in a 5 dimension phase space and the largest Lyapunov exponent dictate us that the maximum length of a prediction should be less than 49 time steps.

Also a simple externally triggered optoelectronic circuit can be used in order first to generate chaotic voltage signals and then to control the obtained chaotic signals by varying specific circuit parameters, for example, the value of a specific component. In the considered optoelectronic circuit, the crucial parameter is the input loop resistor R_1, the value of which leads to the generation of a chaotic voltage signal across the resistor R_1. However, this chaotic voltage reflects a chaotic current through R_1 and, thus, through the optocoupler's input driving LED. Hence, the so driven phototransistor will generate a chaotic voltage signal $\upsilon E(t)$ across emitter resistor R_2, according to the strength of the driving chaotic light signal as controlled by the value of the input loop resistor R_1.

At last a single electron transistor with proper parameters can produce chaotic oscillations, even with the limitations that a simulation states. Chaotic behavior is possible in SET circuits so chaotic systems at nanotechnology is a field for further investigation.

References

1. Hanias, P.M., Tombras, G.S.: Time series cross prediction in a single transistor chaotic circuit. Chaos Solitons Fractals **41**, 1167–1173 (2009)
2. Hanias, P.M., Tombras, G.S., Giannis, I.L.: Chaotic operation by a single transistor circuit in the reverse active region. Chaos **20**, 013105 (2010)
3. Hanias, P.M., Nistazakis, H.E., Tombras, G.S.: Study of an optoelectronic chaotic circuit. 3rd International Interdisciplinary Chaos Symposium on Chaos and Complex Systems, Istanbul, Turkey, 2010
4. Hanias, P.M., Tombras, G.S.: Time series analysis in single transistor chaotic circuit. Chaos Solitons Fractals **40**, 246–256 (2009)
5. Hinrichs, N., Oestreich, M., Popp, K.: Chaos Solitons Fractals **8**(4), 535–558 (1997)
6. Bonakdar, M., Samadia, M., Salarieh, H., Alasty, A.: Chaos Solitons Fractals **36**(3), 682–693 (2008)
7. Bask, M.: Chaos Solitons Fractals **7**(12), 2199–2214 (1996)
8. Halbiniak, Z., Jóźwiak, J.I.: Chaos Solitons Fractals **31**(2), 409–416 (2007)
9. Cooper, C.: Chaos Solitons Fractals **30**(1), 1–10 (2006)
10. Farmer, D., Sidorowich, J.: Phys. Rev. Lett. **59**, 845–848 (1987)
11. Fernandez-Rodriguez, F., Sosvilla-Rivero S., Garcia-Artiles M.: An empirical evaluation of non-linear trading rules. Working paper, no. 16, FEDEA (2001)
12. Shanga, P., Li, X., Kamae, S.: Chaos Solitons Fractals **25**(1), 121–128 (2005)
13. Xiao, Y., Huang, Y.: Chaos Solitons Fractals **20**(3), 497–502 (2004)
14. Akritas, P., Antoniou, I., Ivanov, V.V.: Chaos Solitons Fractals **11**(1–3), 337–344 (2000)
15. Hu, J., Gao, B.J., White, D.K.: Chaos Solitons Fractals **22**(4), 807–819 (2004)
16. Hanias, P.M., Giannaris, G., Spyridakis, A., Rigas, A.: Chaos Solitons Fractals **27**(2), 569–573 (2006)
17. Takens, F.: Dynamical Systems and Bifurcations, Lecture Notes in Mathematics, vol. 1125, pp. 99–106. Springer, New York (1985)
18. Kantz, H., Schreiber, T.: Nonlinear Time Series Analysis. Cambridge University Press, Cambridge (1997)
19. Hong, Y. et al.: Chaos Solitons Fractals **29**(2), 462–469 (2006)
20. Guo-Hui, L.: Chaos Solitons Fractals **26**, 87 (2005)
21. Li, H.G., Zhou, P.S., Yang, K.: Chaos Solitons Fractals **33**, 582 (2007)
22. Abarbanel, I.D.H., Brown, R., Kadtke, B.J.: Phys. Rev. A **41**, 1782 (1990)
23. Cao, L.: Physica D **110**, 43 (1997)
24. Casdagli, M.: Physica D **35**, 335 (1989)
25. Fraser, M.A., Swinney L.: Phys. Rev. A **33**, 1134 (1986)
26. Frede, V., Mazzega, P.: Geophys. J. Int. **137**, 551 (1999)
27. Aasen, T., Kugiumtzis, D.T., Nordahl, G.H.S.: Comput. Biomed. Res. **95** (1997)
28. Rosenstein, T.M., Collins, J.J., Deluca, J.C.: Physica D **65**, 117 (1993)
29. Kennel, B.M., Brown, R., Abarbanel, I.D.H.: Phys. Rev. A **45**, 3403 (1992)
30. Kantz, H., Schreiber, T., Hoffmann, I., Buzug, T., Pfister, G., Flepp, G.C., Simonet, J., Badii, R., Brun, E.: Phys. Rev. E **48**, 1529 (1993)
31. Orzeszko, W.: Chaos Solitons Fractals **38**, 1355 (2008)
32. Karakasidis, E.T., Charakopoulos, A.: Chaos Solitons Fractals **41**, 1723 (2009)
33. Kugiumtzis, D., Lillekjendlie, B., Christophersen, N.: Model. Identif. Contr. **15**, 205 (1994)
34. Hegger, R., Kantz, B.R., Schreiber, T.: Chaos **9**, 413 (1999)
35. Sauer, T.: Time series prediction by using delay coordinate embedding. In: Weigend, A.S., Gershenfeld, N.A. (eds.) Time Series Prediction: Forecasting the Future and Understanding the Past, pp. 175–193. Addison-Wesley, Reading, MA (1994)
36. Packard, H.N., Crutchifield, P.J., Farmer, D.J., Shaw, S.R.: Phys. Rev. Lett. **45**, 712 (2009)
37. Linsay, S.P.: Phys. Lett. A **153**, 353 (1991)
38. Guégan, D., Leroux, J.: Chaos Solitons Fractals **41**, 240 (2009)

39. Stavrinides, G.S., Deliolanis, C.N., Laopoulos, Th., Kyprianidis, M.I., Miliou, N.A., Anagnostopoulos, N.A.: The intermitten behavior of a second-order non-linear nonautonomous oscillator. Chaos Solitons Fractals **36**, 1191–1199 (2008)
40. Lonngren, E.K.: IEEE Trans. Edu. **34**(1), (1991)
41. Ferry, K.D., Khoury, M., Gerousis, C., Rack, J.M., Gunther, A., Goodnick, M.S.: Single-electron charging effects in SiMOS devices. Physica E **9**, 69–75 (2001)
42. Korotkov, N.A.: Intrinsic noise on single – electron transistor. Phys. Rev. B **49**(15), 10381–10392 (1994)
43. Yen, L.J. et al.: J. Fundam. Sci. **1**, 1–6 (2005)
44. Yu, S.Y., Lee, S.H., Hwang, W.S.: J. Korean Phys. Soc. **33**, S269–S272 (1998)
45. Gandhi, G., Roska, T., Csurgay, A.: Single electron transistor based chua type chaotic circuit: A SPICE assisted proof. 8th European Conference on Circuit Theory and Design, 2007, ECCTD 2007. 1, DOI 10.1109/ECCTD.2007.4529751, 934–937 (2007)
46. Mykolaitis, G., Tamaševičius, A., Bumelienė, S.: Electron. Lett. **40**(2), 91–92 (2004)

Part 3
Nonlinear Dynamics in Stochastic Networks

Chapter 5
Synchronous Dynamics over Numerosity-Constrained Stochastic Networks

Nicole Abaid and Maurizio Porfiri

5.1 Characterization of Numerosity-Constrained Networks

Collective behavior in animal groups is a recognizable phenomenon executed by interacting individuals that exhibit coordinated motions [67]. Social species, such as fish, use collective behavior to capitalize on inherent benefits of social life, such as advantages in energy expenditure, foraging capabilities, and predator evasion [49]. Beyond spatial constraints such as distance which limit interactions between individuals, communication underlying collective behavior is mediated by the perceptual capabilities of the species. Perceptual capabilities include vision, vibration sensing, electrical, and chemical signals, and psychological factors such as numerosity, which quantifies a critical limit to the species perception of natural numbers.

In the seminal work in [7], members of a thousand-member starling flock are observed as they avoid the predation of a falcon and remain in a cohesive group. The maneuvers enacted by the flock, such as rapid direction and density changes, are shown to be well described by considering that each bird interacts with a varying set of six or seven nearby individuals rather than with a full set of group mates occupying a fixed physical distance. This phenomenon may be viewed as primitive counting, where groups with cardinality less than the species' numerosity are perceived by an individual as a specific collection, while groups of more than this limit are perceived as an inexact "many". Similar findings, wherein numerosity has been shown to impact the interactions among peers across species, are presented for fish schools in [1, 70] and human pedestrians in [45].

In this section, we imbed the phenomenon of numerosity in a random network model, which we characterize in the context of graph theory. We analytically

N. Abaid • M. Porfiri (✉)
Department of Mechanical and Aerospace Engineering, Polytechnic Institute of New York University, Brooklyn, NY 11201, USA
e-mail: nabaid01@students.poly.edu

S. Banerjee et al. (eds.), *Applications of Chaos and Nonlinear Dynamics in Science and Engineering - Vol. 2*, Understanding Complex Systems,
DOI 10.1007/978-3-642-29329-0_5,

compute measures of the expected structure of this class of graphs based on cycles, clustering, and sorting among vertices and theoretical results are verified with numerical simulation. These results are further used to understand the influence of network parameters on graph structures in arbitrarily large networks and in comparison to undirected Erdős–Rényi random networks.

5.1.1 Network Model

We define a class $\mathcal{G}$ of simple random directed graphs on N vertices called numerosity-constrained (NC) graphs. In general, we consider a directed graph $G = (V, E)$ to be a vertex set $V = \{1, 2, \ldots, N\}$ and an edge set $E = \{(i,j) \in V \times V : \exists$ an edge from i to $j\}$. For an edge $(i,j) \in E$, we call i the origin of the edge and j the terminus of the edge and we do not consider self-loops or multiple edges. We call the vertex subset $\mathcal{N}_i = \{j \in V : (i,j) \in E\}$ the neighbors of i and the cardinality of this set is the out-degree of i, written $d(i)$. The in-degree of i, written as $d^I(i)$, is the cardinality of $\mathcal{N}_i^I = \{j \in V : (j,i) \in E\}$, which is the set of vertices that i is said to neighbor.

We construct $G \in \mathcal{G}$ by fixing the out-degree at a constant $n \in \{0, 1, \ldots, N-1\}$, that is, $d(i) = n, \forall i \in V$, where n is the so-called numerosity constant. For all $i \in V$, the edge set originating at i is constructed by assigning exactly n edges from the set $\{i\} \times V \setminus \{i\}$, where each edge connecting i with a peer vertex is selected with equal probability. Therefore, the set of neighbors of vertex i are independent, identically distributed (IID) random variables with cardinality n.

A graph can be uniquely defined as the Laplacian matrix $L = D - A$, where D and A are called the degree and adjacency matrices, respectively. The degree matrix $D \in \mathbb{R}^{N \times N}$ is the diagonal matrix whose ith diagonal element equals $d(i)$. The adjacency matrix $A \in \mathbb{R}^{N \times N}$ is a matrix whose ijth entry is one if $(i,j) \in E$ and zero otherwise. In the case of a NC graph, $D = nI_N$, where I_N is the N-dimensional identity matrix and A has diagonal entries equal zero and the off-diagonal entries in every row comprising n ones and $N - n - 1$ zeros. Since every peer vertex is equally likely to be selected as an edge terminus, every combination of the off-diagonal entries in the rows of A are equally likely. By construction, L is a realization of an IID matrix with zero row-sum, that is, $L1_N = 0$ where $1_N \in \mathbb{R}^N$ is the vector with all entries equal to one. Figure 5.1 shows two exemplary graphs in $\mathcal{G}$ with $N = 5$ and $n = 2$. The graph Laplacians of the left and right graphs are respectively

$$L = \begin{bmatrix} 2 & 0 & -1 & 0 & -1 \\ 0 & 2 & 0 & -1 & -1 \\ 0 & -1 & 2 & 0 & -1 \\ -1 & -1 & 0 & 2 & 0 \\ 0 & -1 & -1 & 0 & 2 \end{bmatrix} \text{ and } L = \begin{bmatrix} 2 & -1 & 0 & 0 & -1 \\ -1 & 2 & 0 & 0 & -1 \\ -1 & 0 & 2 & 0 & -1 \\ -1 & -1 & 0 & 2 & 0 \\ 0 & -1 & 0 & -1 & 2 \end{bmatrix}. \quad (5.1)$$

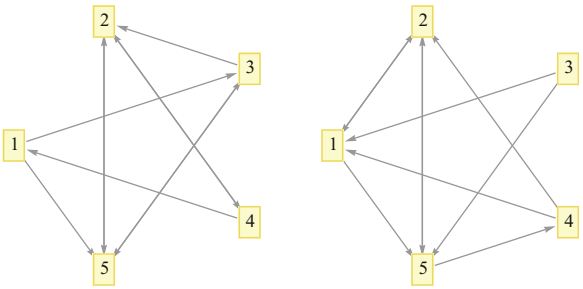

Fig. 5.1 Two realizations of NC graphs with $N = 5$ and $n = 2$. The vertices of the left graph have out-degree sequence $(d(i))_{i=1}^{N} = (2, 2, 2, 2, 2)$ and in-degree sequence $(d^I(i))_{i=1}^{N} = (1, 3, 2, 1, 3)$, and the vertices of the right graph have degree sequences $(d(i))_{i=1}^{N} = (2, 2, 2, 2, 2)$ and $(d^I(i))_{i=1}^{N} = (3, 3, 0, 1, 3)$

5.1.2 Graph Properties

Here, we characterize the expected structure of NC graphs through analytical computation of measures of short-length cycles, clustering, and sorting among vertices.

5.1.2.1 In-degree Distribution

Since every vertex in a NC graph selects exactly n neighbors uniformly from its peer vertices, the out-degree of vertices in $G \in \mathcal{G}$ has a constant distribution and the in-degree has a binomial distribution with parameters $N - 1$ and $n/(N - 1)$, that is,

$$\mathbf{Pr}\left[d^I(i) = k\right] = C_{N-1,k}\left(\frac{n}{N-1}\right)^k \left(1 - \frac{n}{N-1}\right)^{N-1-k}, \tag{5.2}$$

where $k \in \{0, 1, \ldots, N-1\}$ and $C_{(\cdot),(\cdot)}$ is the binomial coefficient. We use the notation $\mathbf{Pr}[\cdot]$, $\mathbf{E}[\cdot]$, and $\mathrm{var}[\cdot]$ for the probability, expected value, and variance of a random variable, respectively. From (5.2), the in-degree of a vertex has $\mathbf{E}\left[d^I\right] = n$ and $\mathrm{var}\left[d^I\right] = n(N - n - 1)/(N - 1)$, which corroborates the intuition that the mean of the in- and out-degree distributions are equal since every edge has both an origin and a terminus. We note that the in-degree sequence has the same distribution as an undirected Erdős–Rényi random network, see [22], due to the uniform choice of neighbors among the vertices. We define an undirected Erdős–Rényi random network on N vertices to be a graph containing each possible bi-directional edge with uniform probability p.

5.1.2.2 Mixing Patterns

The edges of a graph can be thought of as interactions between the vertices at their origins and termini and this pair of vertices can be classified using a variety of attributes. The correlation of vertex-based attributes, introduced in [47], evidences a preference for certain types of interactions at the graph-wide level. We consider the out- and in-degrees of vertices in a directed graph to be a characteristic measuring a vertex's output and intake of information, respectively, and we seek to find correlations between the output and/or intake of vertices which share an edge. Assortative mixing refers to the statistical correlation between high (or low) out-degree origins and in-degree termini sharing an edge (that is, correlation coefficient close to +1) and disassortative mixing refers to the statistical correlation between high (or low, respectively) out-degree origins and low (or high, respectively) in-degree termini sharing an edge (that is, correlation coefficient close to -1), see [47].

Drawing from the elements of the edge set of $G \in \mathcal{G}$, we compute a correlation coefficient between the origin's out-degree and terminus's in-degree. Since every origin has out-degree equal to at least one and every terminus has in-degree equal to at least one, we compute this coefficient using the so-called remaining out- and in-degrees of the vertices For the ith edge in E, the remaining out-degree of the origin, denoted $o(i)$, is given by the origin's out-degree minus one and the remaining in-degree of the terminus, denoted $t(i)$, is given by the terminus's in-degree minus one. Using a measure of the disagreement between two N-element sequences α and β given by $\sigma(\alpha,\beta) = \sum_{i=1}^{N}\alpha(i)\beta(i) - \frac{1}{N}\sum_{i=1}^{N}\alpha(i)\sum_{j=1}^{N}\beta(j)$, the assortativity coefficient $a(G)$ is

$$a(G) = \frac{\sigma(o,t)}{\sigma(o,o)\sigma(t,t)}. \tag{5.3}$$

Since the remaining out-degree is constant over $G \in \mathcal{G}$, $\sum_{i=1}^{nN} o(i) = n(n-1)N$ and $\sum_{i=1}^{nN} o(i)t(i) = (n-1)\sum_{i=1}^{nN} t(i)$. Substituting these equalities into (5.3), we find that $a(G) = 0, \forall G \in \mathcal{G}$. Therefore, NC graphs including those in Fig. 5.1, show no assortative or disassortative mixing among vertex degrees. Other examples of networks with no degree assortativity are Erdős–Rényi random graphs [20, 21] and scale-free networks [8, 46].

5.1.2.3 Cycles

To measure the geometry of connections in a NC graph, we define a path of length m from vertex i to vertex j as a sequence of $m + 1$ distinct vertices starting at i and ending at j where consecutive vertices are adjacent with respect to the edges of the graph. An m-cycle is a path of length m with the same first and last vertex and a cycle is said to be at vertex i if we can write this sequence beginning and ending with i.

We count the expected number of short-length cycles in a graph by first considering the probability of a 2-cycle at vertex i. These graph features indicate

two-way communication between vertices and are thus called undirected links; the total number of undirected links in a graph is written as U. For example, in Fig. 5.1, the left graph has $U = 3$ and the right graph has $U = 2$. In general, an undirected link exists at vertices i and j if $(i, j) \in E$ and $(j, i) \in E$ and $\mathbf{Pr}[(i,j) \in E] = d(i)/(N-1)$ and $\mathbf{Pr}[(j,i) \in E] = d(j)/(N-1)$. Since the out-degrees of vertices in the NC graph are constant, the probability of both these events equals the product $n^2/(N-1)^2$ for $i \neq j$ and the expected number of undirected links at the $N(N-1)/2$ distinct vertex pairs in $G \in \mathcal{G}$ is

$$\mathbf{E}[U] = \frac{Nn^2}{2(N-1)}. \tag{5.4}$$

Since the number of distinct ordered vertex sequences of length m is $N(N-1)\ldots(N-m+1)/m$, the expected number of general m-cycles in G is

$$\mathbf{E}[|\{m-\text{cycles in } G\}|] = \frac{N!}{m(N-m)!}\frac{n^m}{(N-1)^m}. \tag{5.5}$$

5.1.2.4 The Clustering Coefficient

Groupings of three vertices can be investigated through the number of 3-cycles and the clustering coefficient of a graph. The clustering coefficient, introduced in [71] based on the number of "triangles" in an undirected graph, quantifies the existence of subsets of V within which all the vertices are connected. In particular, the clustering coefficient counts the proportion of the neighbors of each vertex which are also connected by an edge. This measure is extended to directed graphs in [14] by neglecting the orientation of the directed edges, that is, by considering the undirected clustering coefficient divided by two. However, by omitting orientation information, this directed clustering coefficient equally counts clusters which are weakly and strongly connected, see [23]. Here, we extend the undirected definition only to strongly connected clusters by redefining the clustering coefficient of a vertex i using the number of 3-cycles at i, which we call directed triangles $T(i)$ [2]. As an analogy to the case of undirected links, the number of directed triangles can be thought of as quantifying the number of edges linking the neighbors of a vertex i and the vertices that i neighbors. The number of directed triangles at i is $T(i) = \sum_{j\in\mathcal{N}_i}\sum_{k\in\mathcal{N}_i'}A_{jk}$, and the total number of directed triangles in a graph is $T = \sum_{i=1}^{N}T(i)/3$ to avoid double-counting. As an illustration, in Fig. 5.1, the left graph has $T = 1$ and the right graph has $T = 3$. We comment that, from (5.5), the expected number of triangles in $G \in \mathcal{G}$ is

$$\mathbf{E}[T] = \mathbf{E}\left[\sum_{i=1}^{N} T(i)\right] = \frac{N(N-2)n^3}{3(N-1)^2} \tag{5.6}$$

The clustering coefficient is the sum of the number of directed triangles normalized by the number of the "potential" directed triangles at each vertex. We determine the number of potential directed triangles as the number of edge pairs $\{(i,j),(k,i)\} \in E \times E$ such that $(i,j) \in \mathcal{N}_i$, $(k,i) \in \mathcal{N}_i^I$, and $j \neq k$. The directed triangles in G correspond to edges of the form (j, k) actualizing these potential triangles, which we combine as The Clustering Coefficient

$$c = \sum_{i=1}^{N} \left(\frac{\sum j \in \mathcal{N}_i, k \in \mathcal{N}_i^I A_{jk}}{|\mathcal{N}_i||\mathcal{N}_i^I| - |\mathcal{N}_i \bigcap \mathcal{N}_i^I|} \right), \tag{5.7}$$

where $\bigcap$ denotes the intersection of sets. As an example, in Fig. 5.1, the left graph has $c = 5/6$ and the right graph has $c = 5/2$.

We use the adjacency matrix to write a closed form expression for the expected clustering coefficient of $G \in \mathcal{G}$. The potential triangles at $i \in G$ correspond to pairs of non-zero entries in A of the form $a_{ij} = 1$ and $a_{ki} = 1$, which are realized as directed triangles by $a_{jk} = 1$. This event is denoted Δ_{ijk} and has probability $\mathbf{Pr}[\Delta_{ijk}] = \mathbf{Pr}[\{a_{jk} = 1\}]$ by the independence of the rows of A. Using a counting argument, $\mathbf{Pr}[\Delta_{ijk}] = C_{N-2,n-1} = n/(N-1)$. Then, the expected clustering coefficient of G is

$$\mathbf{E}[c] = \sum_{i=1}^{N} \mathbf{Pr}[\Delta_{ijk}] = \frac{Nn}{(N-1)}. \tag{5.8}$$

5.1.3 Discussion

We verify analytical results on the expected properties of NC graphs with numerical simulations and discuss their relationship to network parameters. In addition, we adapt these quantities for arbitrarily large networks and compare these results to the undirected Erdős–Rényi random network.

We compare the analytical forms for the expected number of undirected links and directed triangles and the expected clustering coefficient to a numerical simulation of 10 replicate realizations of a NC graph. We numerically calculate the number of 2- and 3-cycles in a graph by noticing that $(A^r)_{ij}$ is exactly the number of paths of length r from i to j. We comment that this interpretation allows repeated vertices within a path, but this scenario does not arise in the case of 2- and 3-cycles due to their short length. Therefore, the total number of undirected links equals the

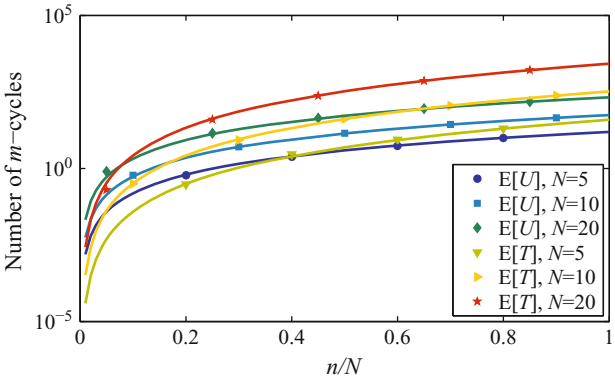

Fig. 5.2 Expected number of undirected links and directed triangles in networks with $N = 5$, 10, and 20 vertices. Theoretical results are shown by *solid curves* and simulation results, averaged over 10 graph realizations, are shown by *markers*

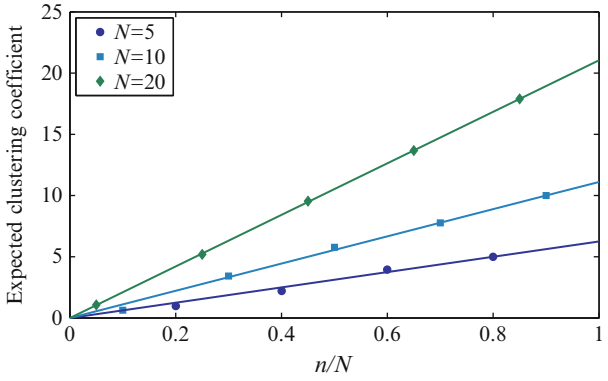

Fig. 5.3 Expected clustering coefficient for an arbitrary vertex in networks with $N = 5$, 10, and 20 vertices. Theoretical results are shown by *solid curves* and simulation results, averaged over 10 graph realizations, are shown by *markers*

trace of $A^2/2$ and the total number of directed triangles equals the trace of $A^3/3$, see [23, 27]. Figure 5.2 shows theoretical and simulation results for $\mathbf{E}[U]$ and $\mathbf{E}[T]$ as n/N is varied for three network sizes. We find that the average number of short-length cycles in $G \in \mathcal{G}$ is well described by the analytical expressions in (5.4) and (5.6). For N fixed, $\mathbf{E}[U] \propto n^2$ and $\mathbf{E}[T] \propto n^3$, which signifies that the local network structure depends exclusively on the numerosity. Figure 5.3 compares the numerical expected clustering coefficient calculated from the mean of 10 realizations of $\mathcal{G}$ with the analytical value in (5.8) for three network sizes. We observe that the relationship between n and $\mathbf{E}[c]$ is linear, and the slope depends only on N.

For arbitrarily large networks, $\mathbf{E}[U] \approx n^2/2$, $\mathbf{E}[T] \approx n^3/3$, and $\mathbf{E}[c] \approx n$; thus, the expected number of short-length cycles and the clustering coefficient depend on n and are independent of N. We comment that $\mathbf{E}[U] \approx \mathbf{E}[d]^2/2$ and $\mathbf{E}[T] \approx \mathbf{E}[d]^3/3$,

using the expected value of the in-degree distribution in (5.2). This asymptotic behavior is similarly exhibited by an Erdős–Rényi network. From [26], the expected number of undirected triangles in an Erdős–Rényi network scales as $(N-1)^3p^3/6$ for large networks. Since an undirected triangle is equivalent to two directed triangles with opposite orientation, the expected number of directed triangles in an Erdős–Rényi network scales as $(N-1)^3p^3/3$. We comment that the expected number of edges of an Erdős–Rényi network is $N(N-1)p$ and $E = Nn$ for a NC network, where $|\cdot|$ denotes the cardinality of a set. We compare these two network types when N is large and the number of edges in the networks are equal, that is, when $n = (N-1)p$. In this case, the NC network has $\mathbf{E}[T] \approx (N-1)^3p^3/3$. Thus, the expected local structure in NC networks is consistent with Erdős–Rényi networks for arbitrarily large N.

5.2 Consensus over Numerosity-Constrained Networks

The salient features of animal grouping have been successfully condensed into consensus problems, see for example [36], wherein coupled agents seek agreement on a quantity of interest by distributed averaging over a network. The topology of the communication network describes the information flow among the agents and thus dictates their ability to achieve consensus. While consensus problems over deterministic networks have received extensive attention in the technical literature, see for example [11, 64], limited research efforts have focused on the analysis of consensus problems over stochastic networks. Criteria for almost sure consensus over undirected Erdős–Rényi networks are presented in [31] and extended to arbitrarily weighted and directed networks in [34, 60, 69, 73], while criteria for mean square consensus over undirected networks are presented in [39]. The convergence rate of consensus protocols is presented in [31, 33, 39, 50] for stochastic undirected networks and in [52, 77] for directed networks. These studies largely focus on stochastic networks where each link is independent of the others.

In this section, we analyze the discrete-time consensus problem for a group of agents that communicate through a stochastically-switching NC network, where links are not independent since vertex out-degree is predetermined. We find necessary and sufficient conditions for mean square consentability of the consensus protocol and we derive a closed form expression for the asymptotic convergence factor. Analytical results are illustrated through simulations.

5.2.1 Problem Statement

We write the discrete-time consensus problem as

$$x(k+1) = (I_N - \varepsilon L(k))x(k), \tag{5.9}$$

where $x(k) \in \mathbb{R}^N$ is the state of N networked agents at time step $k \in \mathbb{Z}^+$, $x(0) \in \mathbb{R}^N$ is the initial condition, and $L(k)$ is the graph Laplacian associated with a dynamically-switching NC network over N agents with numerosity n. The matrices $L(k)$ are IID random matrices with common random variable L. The parameter $\varepsilon>0$ weights the averaging of the state of an agent against its neighbors', which quantifies the propensity of each agent to compromise, see [55]. This parameter can also be viewed as the network switching rate when the consensus problem is written in a continuous-time setting and the system dynamics are discretized using a first order Euler approximation. We note that the entries of $I_N - \varepsilon L(k)$ are not required to be nonnegative, since we do not require ε to be less or equal than $1/n$.

5.2.2 *Preliminary Results*

We rewrite the consensus problem (5.9) as a stochastic stability problem by using the method illustrated in [60]. First, we recall basic notions of stochastic stability and their adaptation to consensus problems.

5.2.2.1 Stochastic Stability

Consider the generic discrete-time linear system

$$x(k+1) = W(k)x(k), \tag{5.10}$$

where $x(0) \in \mathbb{R}^N$ is the initial condition and $W(k)$'s are realizations of IID matrices in $\mathbb{R}^{N\times N}$ with common random variable W.

Definition 1. System (5.10) is mean square stable (MSS) if $\lim_{k\to\infty} \mathbf{E}[\|x(k)\|^2] = 0$ for any $x(0) \in \mathbb{R}^N$.

We note that mean square stability is equivalent to other moment-type stabilities, including exponential mean square stability and stochastic second moment stability and that these equivalent types of stability imply almost sure stability [37].

The asymptotic convergence factor of (5.10), see for example [25], is defined as

$$r_a = \sup_{\|x(0)\|\neq 0} \lim_{k\to\infty} \left(\frac{\mathbf{E}[\|x(k)\|^2]}{\|x(0)\|^2} \right)^{1/k}.$$

We comment that the asymptotic convergence factor is independent of the norm selection due to the equivalence of norms in $\mathbb{R}^N$ and that, for deterministic time-invariant systems, it equals the square of the state matrix's spectral radius. In what follows, we select the Euclidean norm. For stochastic systems, the asymptotic convergence factor can be computed from the statistical properties of W by the following theorem, see for example [3].

Theorem 1. *For system*(5.10), $r_a = \rho(\mathbf{E}[W \otimes W])$ *and* $\rho(\mathbf{E}[W \otimes W]) < 1$ *if and only if* (5.10) *is MSS, where* $\rho(\cdot)$ *denotes the spectral radius of a matrix and* $\otimes$ *denotes the Kronecker product.*

5.2.2.2 Stochastic Consentability

Consensus problems, such as that in (5.9), are characterized by state matrices with the property that 1_N is an equilibrium for the system, that is,

$$W(k)1_N = 1_N. \tag{5.11}$$

To assess the accordance in a system of agents, we define the so-called agreement space $\mathscr{A} = \{v \in \mathbb{R}^N : v = \mu 1_N \text{ with } \mu \in \mathbb{R}\}$ and its orthogonal complement in $\mathbb{R}^N$ as $\mathscr{A}^\perp = \{v \in \mathbb{R}^N : v^{\mathrm{T}}u = 0 \,\forall\, u \in \mathscr{A}\}$, where T is the matrix transpose. Following [60], we project the consensus problem onto $\mathscr{A}^\perp$ by defining a matrix $Q \in \mathbb{R}^{N\times(N-1)}$ with the properties $Q^{\mathrm{T}}1_N = 0$ and $Q^{\mathrm{T}}Q = I_{N-1}$. From the orthogonality of the augmented matrix $[Q \ (1/\sqrt{N})1_N]$, it follows that [77]

$$QQ^{\mathrm{T}} = I_N - \frac{1}{N}1_N 1_N^{\mathrm{T}} := R. \tag{5.12}$$

By construction, R is an orthogonal projection onto $\mathrm{span}\{1_N\}^\perp$ as it is symmetric, idempotent, and $\mathrm{Ker}(R) = \mathrm{span}\{1_N\}$, see [28]. Here, $\mathrm{Ker}(\cdot)$ refers to the kernel of a matrix.

The vector component of $x(k)$ in $\mathscr{A}^\perp$ is $\xi(k) = Q^{\mathrm{T}}x(k) \in \mathbb{R}^{N-1}$, whose norm measures disagreement among the agents in the network. Defining the arithmetic mean value $\bar{x}(k) = \frac{1}{N}1_N^{\mathrm{T}}x(k)$, it follows that $x(k) = \bar{x}(k)1_N + Q\xi(k)$. Thus, the disagreement dynamics is

$$\xi(k+1) = \widetilde{W}(k)\xi(k), \tag{5.13}$$

with $\widetilde{W}(k) = Q^{\mathrm{T}}W(k)Q \in \mathbb{R}^{(N-1)\times(N-1)}$. The disagreement dynamics is used to investigate the stability of disagreement among agents, see for example [52, 77].

Definition 2. System (5.10) with condition (5.11) is mean square consentable if system (5.13) is MSS.

To assess the consentability of the system, the following proposition establishes a manageable expression for the convergence factor of (5.13), which we refer to as the convergence factor of the consensus protocol (5.10).

Proposition 1. *The asymptotic convergence factor of the consensus protocol*(5.10) *is equal to* $\rho((R \otimes R)\mathbf{E}[W \otimes W])$.

The proof, which follows from Theorem 1 and the properties of Q, appears in [3]. We note that the selection of Q is not unique; nonetheless, the asymptotic convergence factor of the consensus protocol is independent of it.

5.2.3 Mean Square Consensus

The convergence factor of the consensus protocol in (5.9) is computed by using $W(k) = I_N - \varepsilon L(k)$ in Proposition 1. Thus, $r_a = \rho(G)$ for

$$G = (R \otimes R)\big(I_{N^2} - \varepsilon(I_N \otimes \mathbf{E}[L] + \mathbf{E}[L] \otimes I_N) + \varepsilon^2 \mathbf{E}[L \otimes L]\big). \tag{5.14}$$

In the following proposition, we compute the matrices $\mathbf{E}[L]$ and $\mathbf{E}[L \otimes L]$ to find G for the consensus protocol over a NC network. For ease of notation, we define $e_i \in \mathbb{R}^N$ to be the vector with ith entry equal to one and all the remaining entries equal to zero.

Proposition 2. *For the consensus protocol over a NC network, the matrix G in* (5.14) *can be written*

$$G = \left(1 - \frac{\varepsilon n N}{N-1}\right)^2 (R \otimes R) + \varepsilon^2 \Delta (I_N \otimes R) F, \tag{5.15}$$

where $\Delta = \frac{n(N-n-1)}{(N-1)(N-2)}$ *and F is the matrix with diagonal and off-diagonal blocks in* $\mathbb{R}^{\mathbb{N}\times\mathbb{N}}$ *equal to*

$$F_{ii} = \frac{1}{N-1}\left(e_i (R e_i)^{\mathrm{T}} - \frac{1}{N} I_N\right) \text{ for } i = 1, \ldots, N \tag{5.16}$$

and

$$F_{ij} = F_{jj} + e_i e_j^{\mathrm{T}} - \frac{1}{N-1} e_i \big(1_N^{\mathrm{T}} - e_i^{\mathrm{T}}\big) \text{ for } i, j = 1, \ldots, N \text{ and } i \neq j, \tag{5.17}$$

respectively. In addition, the matrix G is symmetric.

Proof. We denote the population of all possible distinct Laplacian matrices describing NC graphs on N vertices with numerosity n as $\mathscr{L} = \{L^{(1)}, L^{(2)}, \ldots, L^{(g)}\}$, where g is the number of possible NC graphs. Considering every graph in $\mathscr{L}$ to be equally likely, we calculate $\mathbf{E}[L]$ and $\mathbf{E}[L \otimes L]$ using a counting argument. We write that each row of $L^{(r)}$ has $C_{N-1,\,n}$ possible arrangements throughout the directed graph population. The number of NC graphs is $g = (C_{N-1,\,n})^N$ since the rows of $L^{(r)}$ are independent, and a given off-diagonal entry is nonzero in $C_{N-2,\,n-1}$ $(C_{N-1,\,n})^{N-1}$ of these g matrices. Thus, the diagonal elements of the expected graph Laplacian equal n and the off-diagonal elements equal $-n/(N-1)$; the NC graph Laplacian is then

$$\mathbf{E}[L] = \frac{nN}{N-1} R. \tag{5.18}$$

Calculating $\mathbf{E}[L \otimes L]$ requires formulation of the structure of the generic matrix $L^{(r)} \otimes L^{(r)}$ for $r = 1, \ldots, g$. The entries of this matrix belong to the set $\{0, 1, -n, n^2\}$ and, for convenience, we group the generic term of this matrix $L_{ij}L_{st}$ occupying in the stth position in the ijth block into six disjoint cases: (1) $i = j$, $s = t$; (2) $i \neq j$, $i = s$, $j = t$; (3) $i = j$, $s \neq t$; (4) $i \neq j$, $s = t$; (5) $i \neq j$, $i = s$, $j \neq t$, $s \neq t$; and (6) $i \neq j$, $i \neq s$, $s \neq t$. The expected values of six cases, denoted $\alpha_1, \ldots, \alpha_6$, are calculated using a counting argument. We find that $\alpha_1 = n^2$, $\alpha_2 = n/(N-1)$, $\alpha_3 = \alpha_4 = -n^2/(N-1)$, $\alpha_5 = n(n-1)/((N-1)(N-2))$, and $\alpha_6 = n^2/(N-1)^2$, see for example [3]. As an example of the block structure of this matrix, for $N = 3$, $\mathbf{E}[L \otimes L]$ equals

$$\begin{bmatrix}
\alpha_1 & \alpha_3 & \alpha_3 & \alpha_4 & \alpha_2 & \alpha_5 & \alpha_4 & \alpha_5 & \alpha_2 \\
\alpha_3 & \alpha_1 & \alpha_3 & \alpha_6 & \alpha_4 & \alpha_6 & \alpha_6 & \alpha_4 & \alpha_6 \\
\alpha_3 & \alpha_3 & \alpha_1 & \alpha_6 & \alpha_6 & \alpha_4 & \alpha_6 & \alpha_6 & \alpha_4 \\
\alpha_4 & \alpha_6 & \alpha_6 & \alpha_1 & \alpha_3 & \alpha_3 & \alpha_4 & \alpha_6 & \alpha_6 \\
\alpha_2 & \alpha_4 & \alpha_5 & \alpha_3 & \alpha_1 & \alpha_3 & \alpha_5 & \alpha_4 & \alpha_2 \\
\alpha_6 & \alpha_6 & \alpha_4 & \alpha_3 & \alpha_3 & \alpha_1 & \alpha_6 & \alpha_6 & \alpha_4 \\
\alpha_4 & \alpha_6 & \alpha_6 & \alpha_4 & \alpha_6 & \alpha_6 & \alpha_1 & \alpha_3 & \alpha_3 \\
\alpha_6 & \alpha_4 & \alpha_6 & \alpha_6 & \alpha_4 & \alpha_6 & \alpha_3 & \alpha_1 & \alpha_3 \\
\alpha_2 & \alpha_5 & \alpha_4 & \alpha_5 & \alpha_2 & \alpha_4 & \alpha_3 & \alpha_3 & \alpha_1
\end{bmatrix}.$$

For arbitrary N, $\mathbf{E}[L \otimes L]$ has diagonal and off-diagonal blocks equal to

$$\mathbf{E}[L \otimes L]_{ii} = \frac{n^2 N}{N-1} R, \quad \text{for } i = 1, \ldots, N \tag{5.19}$$

and

$$\mathbf{E}[L \otimes L]_{ij} = -\frac{n^2 N}{(N-1)^2} R + \Delta e_i e_j^{\mathrm{T}} - \frac{\Delta}{N-1} e_i \left(1_N^{\mathrm{T}} - e_i^{\mathrm{T}}\right), \quad \text{for } i, j = 1, \ldots, N, \; i \neq j, \tag{5.20}$$

respectively with $\Delta = \frac{n(N-n-1)}{(N-1)(N-2)}$. The claim in (5.15) follows from substitution of (5.18)–(5.20) into (5.14). The symmetry of $(I_N \otimes R)F$ is verified through direct computation, thus yielding the symmetry of G. □

Equation (5.15) shows that the effect of the parameters n and ϵ on G resides in two nonnegative constants weighting the linear combination of the matrices $R \otimes R$ and $(I_N \otimes R)F$. Computing $\rho(G)$ requires the following result relating the eigenvectors and eigenvalues of G and $(I_N \otimes R)F$.

Proposition 3. *Define the $2N - 1$ dimensional subspace $\Gamma^{(1)} = \{v \in \mathbb{R}^{N^2} : v = w \otimes 1_N$ or $v = 1_N \otimes w$ where $w \in \mathbb{R}^N\}$ and its orthogonal complement $\left(\Gamma^{(1)}\right)^{\perp}$. Considering the matrices in Proposition 2, $\Gamma^{(1)}$ is a subspace of the null spaces*

of $(I_N \otimes R)F$ and G. In addition, if $v \in \left(\Gamma^{(1)}\right)^{\perp}$ is an eigenvector of $(I_N \otimes R)F$ with eigenvalue $\lambda \in \mathbb{R}$, then v is an eigenvector of G with eigenvalue $\left(1 - \frac{\varepsilon nN}{N-1}\right)^2 + \varepsilon^2 \Delta\lambda$.

The proof of this proposition appears in [3]. The main result on consensus over NC networks follows.

Theorem 2. *The asymptotic convergence factor of the consensus protocol in (5.9) with G as in (5.15) is*

$$r_a = \rho(G) = 1 - \varepsilon \frac{2nN}{N-1} + \varepsilon^2 \left(n^2 + n + \frac{n^2 N}{(N-1)^2} \right). \tag{5.21}$$

Proof. The theorem is demonstrated by identifying the eigenvalues and eigenvectors of $(I_N \otimes R)F$, which determine the spectrum of G, and computing the spectral radius directly from comparison of the eigenvalues' magnitudes.

From Proposition 3, $\Gamma^{(1)}$ is in the null space of G and contains an eigenspace of G with eigenvalue $\lambda^{(1)} = 0$. We compute the remaining eigenvalues of G whose eigenvectors are in $\left(\Gamma^{(1)}\right)^{\perp}$ by appealing to Proposition 3. Using the eigenvalue equation, it can be verified that $\widehat{\lambda}^{(2)} = -(N-2)/(N-1)$, $\widehat{\lambda}^{(3)} = 0$, and $\widehat{\lambda}^{(4)} = N-2$ are eigenvalues of $(I_N \otimes R)F$ whose corresponding eigenspaces, with eigenvectors $v = \left[v_1^{\mathrm{T}} \ \ldots v_N^{\mathrm{T}} \right]^{\mathrm{T}}$, contain the respective subspaces.

$$\Gamma^{(2)} = \left\{ v \in \mathbb{R}^{N^2} : v_i = \mu_i R e_i - \frac{1}{N} \sum_{j=1}^{N} \mu_j R e_j, \text{with } \mu_i \in \mathbb{R}, \sum_{j=1}^{N} \mu_j = 0 \text{ for } i = 1, \ldots, N \right\},$$

$$\Gamma^{(3)} = \left\{ v \in \mathbb{R}^{N^2} : \sum_{i=1}^{N} v_i = 0, v_i^{\mathrm{T}} 1_N = 0, \text{and } e_i^{\mathrm{T}} R v_i = 0 \text{ for } i = 1, \ldots, N \right\},$$

$$\Gamma^{(4)} = \left\{ v \in \mathbb{R}^{N^2} : v_i = \mu R e_i \text{ with } \mu \in \mathbb{R} \text{ for } i = 1, \ldots, N \right\}.$$

In addition, $\Gamma^{(2)}$, $\Gamma^{(3)}$, and $\Gamma^{(4)}$ are orthogonal to $\Gamma^{(1)}$, which is directly verifiable. Therefore, Proposition 3 yields that

$$\lambda^{(2)} = 1 - \varepsilon \frac{2nN}{N-1} + \varepsilon^2 \left(\frac{n^2(N^2+1)}{(N-1)^2} - \frac{n}{N-1} \right),$$

$$\lambda^{(3)} = \left(1 - \frac{\varepsilon nN}{N-1} \right)^2,$$

$$\lambda^{(4)} = 1 - \varepsilon \frac{2nN}{N-1} + \varepsilon$$

are eigenvalues of G with corresponding eigenspaces containing $\Gamma^{(2)}$, $\Gamma^{(3)}$, and $\Gamma^{(4)}$, respectively. Note that $\lambda^{(3)}$ and $\lambda^{(4)}$ are always nonnegative, while $\lambda^{(2)}$ may change the sign as ε and n are varied.

It can be verified that $\Gamma^{(2)}$, $\Gamma^{(3)}$, and $\Gamma^{(4)}$ are mutually orthogonal and their dimensions are $N-1$, N^2-3N+1, and 1, respectively. Since the dimension of $\Gamma^{(1)}$ is $2N-1$, the direct sum of $\Gamma^{(i)}$, $i=1,\ldots,4$ is $\mathbb{R}^{N^2}$. Therefore, $\{\lambda^{(1)}, \lambda^{(2)}, \lambda^{(3)}, \lambda^{(4)}\}$ comprise the spectrum of G. Since $\lambda^{(4)} \geq \lambda^{(3)}$, $\lambda^{(4)} + \lambda^{(2)} \geq 0$, and $\lambda^{(4)} - \lambda^{(2)} \geq 0$, the spectral radius of G is $\lambda^{(4)}$, that yields the claim by Proposition 1.

For a given n, the asymptotic convergence factor equals 1 at $\varepsilon = 0$, has negative derivative in ε at $\varepsilon = 0$, attains a minimum, and increases above unity for sufficiently large ε. In other words, mean square consentability is only possible for sufficiently small values of the ε, which controls the convergence speed.

We comment that, for small values of ε, second order terms in ε can be neglected in (5.14) and the network can be assimilated to a complete graph, whose graph Laplacian appears in (5.18), which supports consensus. As ε increases, agents become more persuasible to rapid state variations. In particular, mean square consentability is possible if $\varepsilon < \overline{\varepsilon} = \frac{2N(N-1)}{nN+(n+1)(N-1)^2}$ and $\overline{\varepsilon}/2$ maximizes the convergence speed. The associated optimal asymptotic convergence factor is $\overline{\varepsilon}(1-(n+1)/N)/2$.

For large N and fixed n, the convergence factor in (5.21) approaches $1-2\varepsilon n+\varepsilon^2(n^2+n)$ and the critical value of ε that admits mean square consensus approaches $2/(n+1)$. The optimal value of ε is $1/(n+1)$ with convergence factor equal to $1/(n+1)$. This value of ε corresponds to a distributed averaging process in which each agent weights its own state the same as its neighbors' states.

In light of the topological analysis in Sect. 5.1, these findings indicate that the network consentability is controlled by the expected clustering coefficient, which for large networks is equal to the numerosity n. Notably, as n increases, the range of values of ε for which consensus is possible becomes narrower. At the same time, increasing ε allows for achieving faster convergence rates.

5.2.4 Simulation Results

We validate and illustrate these results by comparing the NC network with a deterministic lattice on N vertices where each vertex has in- and out-degree equal to n. The corresponding Laplacian matrix L_{lat} is a circulant matrix whose first row is the vector $[l_0 \quad l_1 \quad \ldots \quad l_{N-1}]$ where $l_0 = n$, $l_1 = l_2 = \ldots = l_n = -1$ and $l_{n+1} = l_{n+2} = \ldots = l_{N-1} = 0$. The matrix $R(I_N - \epsilon L_{\text{lat}})$ is also circulant and it is fully described by its first row. Following [10], we express the eigenvalues of $R(I_N - \varepsilon L_{\text{lat}})$ as a polynomial in $e^{2\pi \mathrm{I}/N}$ whose coefficients are functions of l_i, $i=0,\ldots,N-1$, with I being the imaginary unit. Hence, $\lambda_1 = 0$ and

$$\lambda_{m+1} = 1 - \epsilon\left(n+1-e^{\pi n m \mathrm{I}/N}\left(\frac{\sin(\pi m(n+1)/N)}{\sin(\pi m/N)}\right)\right), \tag{5.22}$$

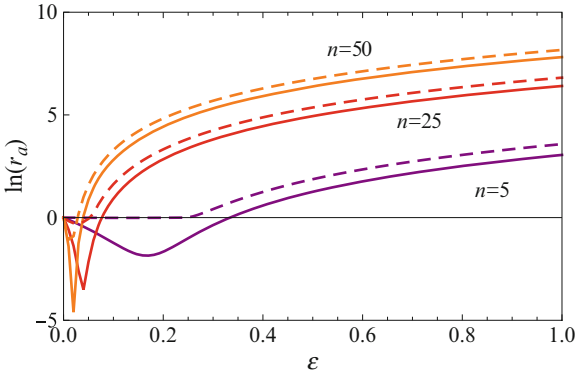

Fig. 5.4 Logarithm of the asymptotic convergence factor for the stochastic NC network and the lattice as a function of ε for $N = 100$ and $n = 5$, $n = 25$, $n = 50$. The *solid line* refers to the stochastic network and the *dashed line* refers to the lattice

for $m = 1, \ldots, N-1$ using identities from [6]. Thus, the asymptotic convergence factor of the consensus protocol over the lattice is $\rho(R(I_N - \epsilon L_{\text{lat}}))^2$.

Figure 5.4 shows the asymptotic convergence factor as a function of ε with $N = 100$, n varying, and using both the stochastic NC and fixed lattice networks. For all selected values of the numerosity, the NC network outperforms the lattice, which may be attributed to increased information mixing over the NC network. As n approaches $N - 1$, both networks go to the complete graph and the performance difference between them decreases. We notice that Fig. 5.4 also illustrates that the range of values of ε that guarantees mean square consentability for NC networks generally encompasses scenarios where ε is larger than $1/n$, which allows negative entries in the state matrix W.

For large networks and fixed n, λ_2 in (5.22) approaches one irrespective of ε, which implies that the convergence factor for protocols over the lattice is bounded from below by a number that approaches one. This implies that the speed of a converging consensus protocol over the lattice approaches zero, in contrast with the consensus protocol executed over the stochastic NC network, whose speed can be tailored by selecting ε.

To demonstrate the dynamics of the consensus protocol over a NC network, we conduct a numerical simulation with $N = 100$, $n = 5$, $\epsilon = 0.2$, and random initial condition selected uniformly in [0, 1]. The first four realizations of the NC communication network are given in Fig. 5.5. We generate simulation data for the consensus protocol over this switching network and we report the agents' states in Fig. 5.6. For this parameter selection, the asymptotic convergence factor in (5.21) is computed as 0.190. Since r_a is less than one, consensus among agents is attainable, as is demonstrated by the state trajectories in Fig. 5.6. We comment that, in general, the realizations of the directed NC network need not be connected in either the weak or strong sense [74].

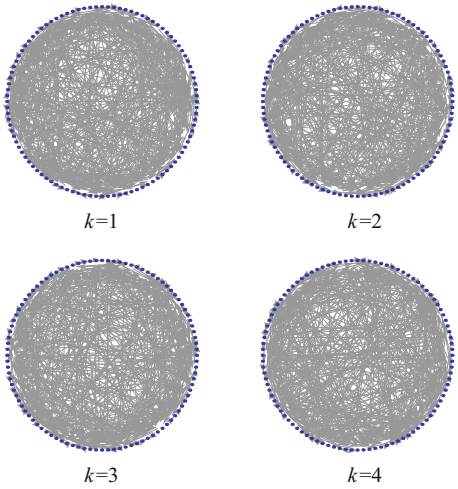

Fig. 5.5 Four realizations of a NC network with $N = 100$ and $n = 5$ indexed by time step k

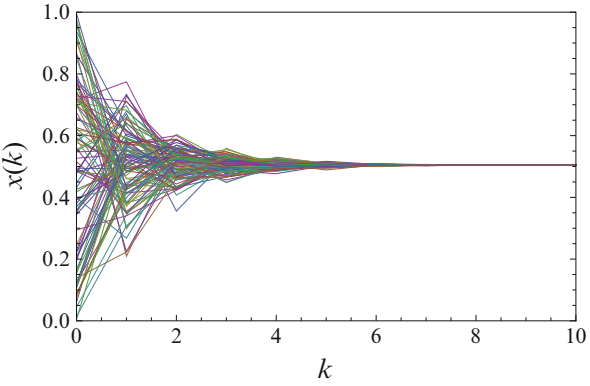

Fig. 5.6 Sample state vector of 100 consensus-seeking agents with $n = 5$ and $\varepsilon = 0.2$

As a validation of the closed form results, Fig. 5.7 presents 25 Monte Carlo simulations over the stochastically switching NC network and lattice with $N = 100$, $n = 5$, and $\varepsilon = 0.2$. After an initial transient, the magnitude of the disagreement vector decreases linearly on a logarithmic scale for both networks. Computing a best fit line over time steps [10, 40] for the NC network type and [270, 300] for the lattice, we find that the square of the disagreement norm decreases as 1.67^k for the NC network and 0.0080^k for the lattice. From (5.21) and (5.22), the analytical convergence factors are computed as 1.66 and 0.0079 respectively, thus confirming the theoretical findings.

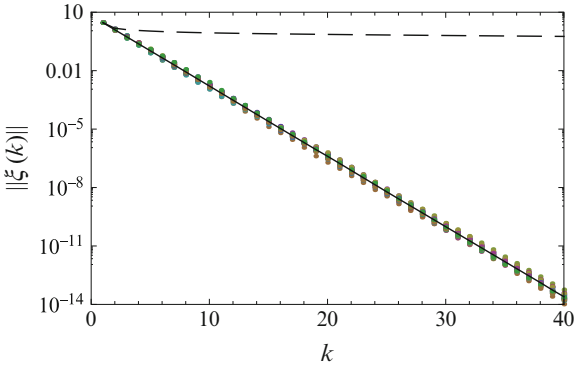

Fig. 5.7 Time evolution of the magnitude of the disagreement vector for 25 Monte Carlo simulations with constant initial conditions and $N = 100$, $n = 5$, and $\varepsilon = 0.2$. The *solid line* shows the average disagreement over the different realizations of the consensus protocol over a switching NC network, depicted as markers. The *dashed line* shows the disagreement in the lattice

5.3 Synchronization over Numerosity-Constrained Networks

Synchronization is a ubiquitous phenomenon that has been observed in a wide variety of systems beyond animal grouping [67] and including fireflies' blinking [13], animal gaits [15], heart stimulation [30], epidemiology [32], and neural activity [72], secure communications [17], chemistry [40], meteorology [19], and optoelectronics [75]. As a result, this phenomenon has been the subject of considerable research efforts in a variety of disciplines. However, the great majority of synchronization research has been focused on dynamical systems that are coupled via static networks whose topology and coupling strengths are constant in time [5, 12, 16, 29, 53]. Within the study of these static network models, the approach based on the master stability function (MSF) introduced by [51] has emerged as an effective tool for studying synchronization. The MSF allows for assessing the linear stability of the synchronization manifold for any interconnecting topology using the spectral properties of the coupling matrix, that can be nondiagonalizable in general [38, 48].

Synchronization over dynamic networks is a relatively untapped field that only recently has attracted significant research efforts, although important applications in modeling complex systems and engineering adaptive networks [18] necessitate such structures. Synchronization over time-varying networks are considered in [4, 35, 42–44, 66, 76] for deterministic network topologies and [9, 61, 62] for stochastic topologies.

In the following section, we study the synchronization of identical chaotic maps coupled by a stochastic NC network. We present a handleable criterion for synchronization based on spectral properties of the coupling network. We illustrate the results through simulations on chaotic logistic and Henon maps coupled through a NC network.

5.3.1 Problem Statement

We consider the synchronization of N chaotic maps whose individual dynamics is governed by $x(k+1) = f(x(k))$ where $x \in \mathbb{R}^m$ is the oscillator state, $f : \mathbb{R}^m \to \mathbb{R}^m$ is a nonlinear function describing the system dynamics, and $k \in \mathbb{Z}^+$ is the time variable. The oscillators are coupled through the graph Laplacians of a stochastic NC network $L(k) \in \mathbb{R}^{N\times N}$ according to the weighting parameter $\varepsilon > 0$. The equations of motion read

$$x_i(k+1) = f(x_i(k)) - \varepsilon \sum_{j=1}^{N} L_{ij}(k) f(x_j(k)), \tag{5.23}$$

where $i = 1, \ldots, N$. We comment that, in the considered scenario, the nonlinear function f dictates the individual dynamics and the inner coupling among interacting oscillators, even if this assumption can be easily released following the arguments presented in [54].

5.3.2 Stochastic Stability of the Synchronization Manifold

We say that the system of oscillators is synchronized if the state vectors for all oscillators are identical, that is, $x_1(k) = \ldots = x_N(k) = s(k)$ for all $k \in \mathbb{N}$ and s is a solution of the individual oscillator dynamics. To study the synchronization of (5.23), we linearize the equations of motion in the neighborhood of $s(k)$ to obtain the variational equations

$$\xi_i(k+1) = Df(s(k))\xi_i(k) - \varepsilon \sum_{j=1}^{N} L_{ij}(k) Df(s(k))\xi_j(k), \tag{5.24}$$

where $i = 1, \ldots, N$, $\xi_i = x_i - s$ is the variation of the ith oscillator, and Df is the Jacobian of f. We decompose the variation of the ith oscillator into a component along the synchronization manifold and a component transverse to the synchronization manifold as

$$\xi_i = \widetilde{\xi}_i + \frac{1}{N} \sum_{j=1}^{N} \xi_j. \tag{5.25}$$

By substituting (5.25) into (5.24) and introducing $\widetilde{\xi}(k) = \left[\widetilde{\xi}_{1(k)^{\mathrm{T}}}, \ldots, \widetilde{\xi}_{N(k)^{\mathrm{T}}}\right]^{\mathrm{T}} \in \mathbb{R}^{mN}$, we find the transverse dynamics to be

$$\widetilde{\xi}(k+1) = (R(I_N - \epsilon L(k)) \otimes Df(s(k)))\widetilde{\xi}(k), \tag{5.26}$$

where R is as in (5.12). The matrix $\varepsilon RL(k)$ can be considered as a modified graph Laplacian following the terminology in [68].

The stochastic stability of the synchronization manifold can be studied by introducing the autocorrelation matrix $\widetilde{\Xi} : \mathbb{N} \to \mathbb{R}^{mN \times mN}$ defined by $\widetilde{\Xi}(k) = \mathbf{E}\left[\widetilde{\xi}(k)\widetilde{\xi}(k)^{\mathrm{T}}\right]$, where initial conditions for the transverse dynamics are considered as parameters. By definition, $\widetilde{\Xi}(k)$ is symmetric and positive semidefinite and its trace quantifies the lack of synchronization, that is, the expected value of $\| \widetilde{\xi}(k)\|^2$. Since the matrices $L(k)$'s are independent random variables, the evolution of $\widetilde{\Xi}$ is given by the recursion [41]

$$\begin{aligned}\mathrm{vec}(\widetilde{\Xi}(k+1)) = &(R \otimes Df \otimes R \otimes Df - \epsilon(R \otimes Df \otimes \mathbf{E}[RL] \otimes Df \\ &+ \mathbf{E}[RL] \otimes Df \otimes R \otimes Df) + \epsilon^2 \mathbf{E}[RL \otimes Df \otimes RL \otimes Df])\mathrm{vec}(\widetilde{\Xi}(k)),\end{aligned} \tag{5.27}$$

where vec denotes vectorization and we have omitted the dependence of the Jacobian on time.

We say that the coupled oscillators stochastically synchronize if (5.27) is stable; on the other hand, if (5.27) is not stable, the oscillators do not stochastically synchronize. This notion of synchronization is based on the concept of mean square stability of stochastic systems [24] and the overarching linearization of the system dynamics.

5.3.3 Stochastic Master Stability Function

We define $v_1, \ldots, v_{N^2}$ as the orthogonal and normalized eigenvectors of G in (5.14) ordered so that the first $2N - 1$ vectors are in $\Gamma^{(1)}$ and correspond to the zero eigenvalue. We denote the eigenvalue corresponding to v_j with $\lambda_j \in \{\lambda^{(2)}, \lambda^{(3)}, \lambda^{(4)}\}$ for $j = 2N, \ldots, N^2$. We diagonalize the variational equation (5.27) by using the decomposition

$$\widetilde{\Xi}(k) = \sum_{j=1}^{N^2} V_j \otimes \Theta_j(k), \tag{5.28}$$

where $V_j = \mathrm{vec}^{-1}(v_j)$ and $\Theta_j : \mathbb{R} \to \mathbb{R}^{m \times m}$ defines the components of $\widetilde{\Xi}$ along v_j at time k for $j = 1, \ldots, N^2$. Therefore, the $p\,q$th component of the matrix $\Theta_j(k)$ equals the trace of the matrix $\widetilde{\Xi}(k)^{\mathrm{T}}\left(V_j \otimes \ e_p e_q^{\mathrm{T}}\right)$.

We substitute the decomposition (5.28) into (5.27) and apply the matrix form operator vec^{-1} to obtain a set of uncoupled equations for Θ_j with $j = 1, \ldots, N^2$ and find

$$\begin{aligned}\sum_{j=1}^{N^2} V_j \otimes \Theta_j(k+1) = &\sum_{j=1}^{N^2} \Big(RV_jR^{\mathrm{T}} \otimes Df\Theta_j(k)Df^{\mathrm{T}} - \epsilon\Big(RV_j\mathbf{E}[RL]^{\mathrm{T}} \otimes Df\Theta_j(k)Df^{\mathrm{T}} \\ &+ \mathbf{E}[RL]V_jR^{\mathrm{T}} \otimes Df\Theta_j(k)Df^{\mathrm{T}}) + \epsilon^2 \mathbf{E}[RLV_j(RL)^{\mathrm{T}}] \otimes Df\Theta_j(k)Df^{\mathrm{T}}),\end{aligned} \tag{5.29}$$

since $(A \otimes B)\text{vec}(C) = \text{vec}\left(BCA^{\text{T}}\right)$ with A, B, and C appropriately sized matrices [10]. By grouping the right hand side of (5.29), we write $\sum_{j=1}^{N^2} V_j \otimes \Theta_j(k+1) = \sum_{j=1}^{N^2} \text{vec}^{-1}(Gv_j) \otimes Df\Theta_j(k)Df^T$. Therefore, by recalling that v_j is an eigenvector of G, we obtain that $\Theta_j(k) = 0$ for $j = 1, \ldots, 2N-1$ and

$$\text{vec}(\Theta_j(k+1)) = \lambda_j(Df \otimes Df)\text{vec}(\Theta_j(k)) \tag{5.30}$$

for $j = 2N, \ldots, N^2$. From (5.30), we define the stochastic master stability equation for the NC-networked dynamical systems

$$\theta(k+1) = \lambda_\star(Df \otimes Df)\theta(k), \tag{5.31}$$

where $\theta \in \mathbb{R}^{m^2}$ is the state of an auxiliary dynamical system and $\lambda_\star$ is the free parameter. System (5.23) is stochastically stable if the largest Lyapunov exponent of (5.31) is negative for $\lambda_\star = \lambda^{(i)}$ with $i = 2, 3, 4$, and is unstable otherwise. The m^2 Lyapunov exponents of (5.31) are $\{\ln(|\lambda_\star|) + h_i + h_j : i = 1, \ldots, m; j = 1, \ldots, m\}$ where $h_1, \ldots, h_m$ are the Lyapunov exponents for the individual map. Therefore, the condition for stochastic synchronization is

$$\ln(\rho(G)) + 2h_{\max} < 0 \tag{5.32}$$

where $\rho(G)$ is as in (5.21) and $h_{\max}$ is the largest Lyapunov exponent of the individual system. We comment that this result extends the classical results on coupled map lattices [63, 65] to stochastically coupled maps. We also note that, even if the network supports stochastic consensus in the sense of $\rho(G)<1$, stochastic synchronization of the system in (5.23) may be not possible due to the inherent chaotic dynamics that require the consensus protocol to overcome the detrimental effect of the chaotic dynamics, that is, $\rho(G)<1/\exp(2h_{\max})$.

Consistent with the previous discussion on consensus for small values of ε, synchronization problems are also practically controlled by the expected coupling matrix for sufficiently small values of ε [56–59, 61, 62, 66].

5.3.4 *Illustration of the Method*

We consider two cases of a NC network of coupled chaotic maps, where the individual systems are either logistic or canonical Henon maps. The dynamics of each individual logistic map is governed by

$$x(k+1) = 3.9x(k)(1 - x(k)), \tag{5.33}$$

and the individual Henon maps are governed by

$$x_1(k+1) = 1 - 1.4x_1^2(k) + x_2(k), \tag{5.34}$$

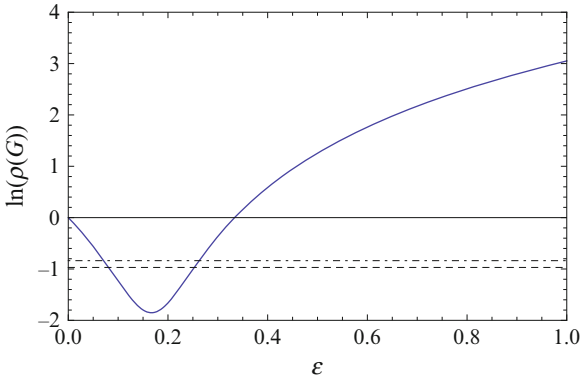

Fig. 5.8 Logarithm of the spectral radius of G for a NC network with $N = 100$ and $n = 5$ as a function of ε (*solid line*) and (-2) times the maximum Lyapunov exponent of the considered chaotic maps (logistic: *dashed line* and Henon: *dash-dotted line*). Synchronization is possible for values of ϵ for which the *solid line* is beneath the *dashed or dash-dotted lines* for logistic or Henon maps, respectively

$$x_2(k+1) = 0.3x_1(k), \tag{5.35}$$

see [65]. The largest Lyapunov exponent $h_{\max}$ is 0.485 for the logistic map and is 0.419 for the Henon map [65]. Therefore, stochastic synchronization is attained if and only if (5.32) holds with the logarithm of the spectral radius of G smaller than -0.970 and -0.838 for logistic and Henon maps, respectively.

We consider a network of 100 oscillators coupled according to a NC network with $n = 5$. Figure 5.8 reports $\ln(\rho(G))$ varying with ε, along with the numerical values of the threshold required for synchronization for both logistic and Henon maps. We observe a bounded interval within which the system is synchronizable, which is equal to [0.09, 0.25] for the logistic maps and equal to [0.08, 0.26] for the Henon maps. Note that the region for which consensus is possible corresponds instead to [0, 0.33], which is considerably wider than the range of ϵ required for the synchronization of the considered chaotic maps.

To demonstrate the dynamics of synchronization, we generate simulation data for 100 logistic maps and 100 Henon maps, coupled in each case according to a NC network with $n = 5$ and various coupling strengths. Random initial conditions are uniformly selected in [0, 0.001] in each case. From Fig. 5.9, we observe that the scalar states for the logistic maps and the first coordinate of the states for the Henon maps synchronize almost immediately when $\varepsilon = 0.2$. This result is consistent with the predictions from (5.32), from which $\ln(\rho(G)) + 2h_{\max}$ equals -0.691 for the logistic maps and -0.823 for the Henon maps. However, when $\varepsilon = 0.05$ and $\varepsilon = 0.33$, the systems fail to synchronize, keeping bounded mismatches between systems in the case of $\varepsilon = 0.05$ or growing unbounded in the

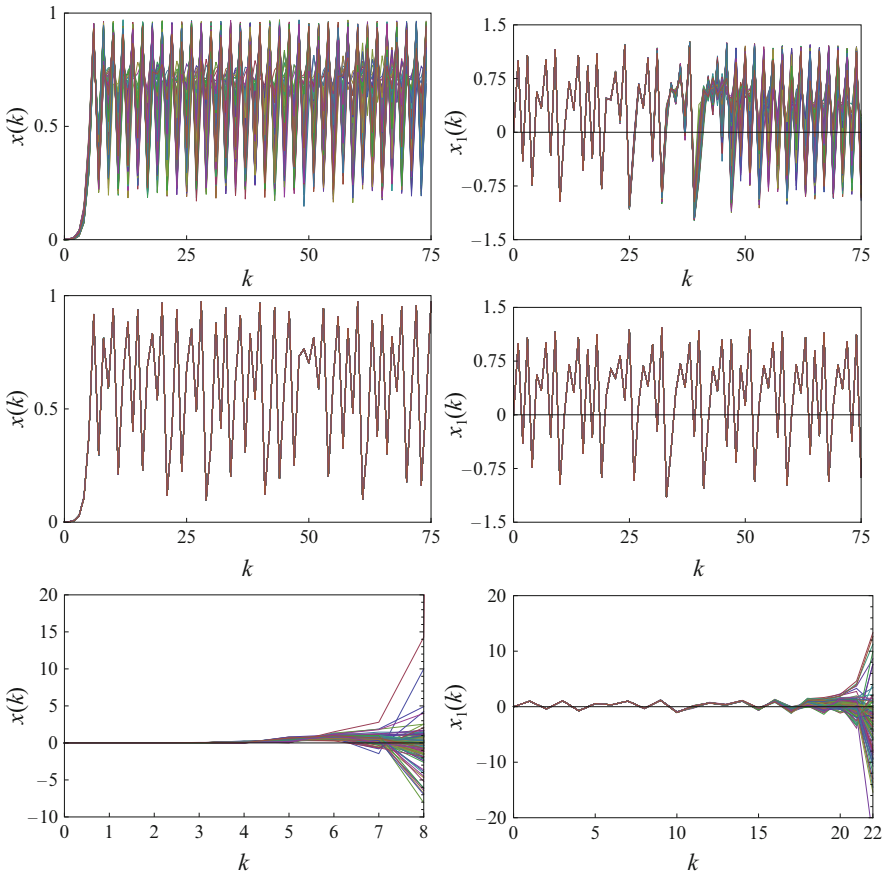

Fig. 5.9 Example of the state vector of 100 logistic maps (*left*) and the first state coordinate of 100 Henon maps (*right*) coupled over a NC network with $n = 5$ and $\varepsilon = 0.05$ (*top*), $\varepsilon = 0.2$ (*middle*), and $\varepsilon = 0.33$ (*bottom*)

case of $\varepsilon = 0.33$. This result is consistent with (5.32), since $\ln(\rho(G)) + 2h_{\max}$ equals 0.409 and 0.931 for the logistic map and 0.277 and 0.799 for the Henon maps, for $\varepsilon = 0.05$ and $\varepsilon = 0.33$ respectively.

To further illustrate the effectiveness of the proposed approach in assessing the network's stochastic synchronization, Fig. 5.10 presents the time evolution of the norm of the error $\delta(k) = \|(R \otimes I_m)x(k)\|$ for the same simulations considered in Fig. 5.9, where $x(k) = \left[x_{1(k)}^{\mathrm{T}}, \ldots, x_{N(k)}^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{mN}$. This error quantifies the mismatch between the oscillators' states for a given trajectory and, when the individual oscillators are proximal to the synchronization manifold, its expectation equals the norm of $\xi(k)$. We consider the simulation set presented in Fig. 5.9 for logistic and Henon maps using three values of the coupling strength. In line with

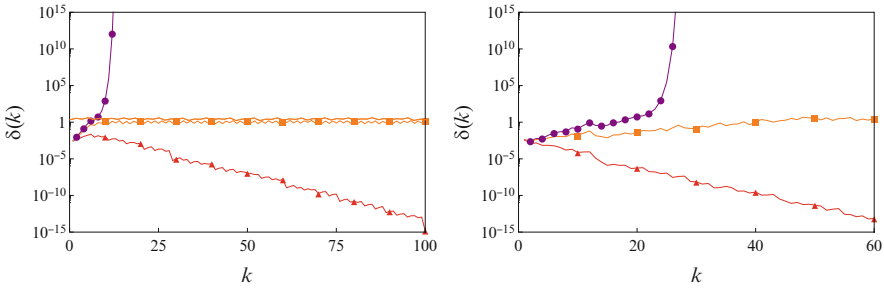

Fig. 5.10 Time evolution of the synchronization error for networks of $N = 100$ logistic maps (*left*) and $N = 100$ canonical Henon maps (*right*) coupled via a NC network with $n = 5$ and with ε equal to 0.05 (*squares*), 0.2 (*triangles*), and 0.33 (*circles*)

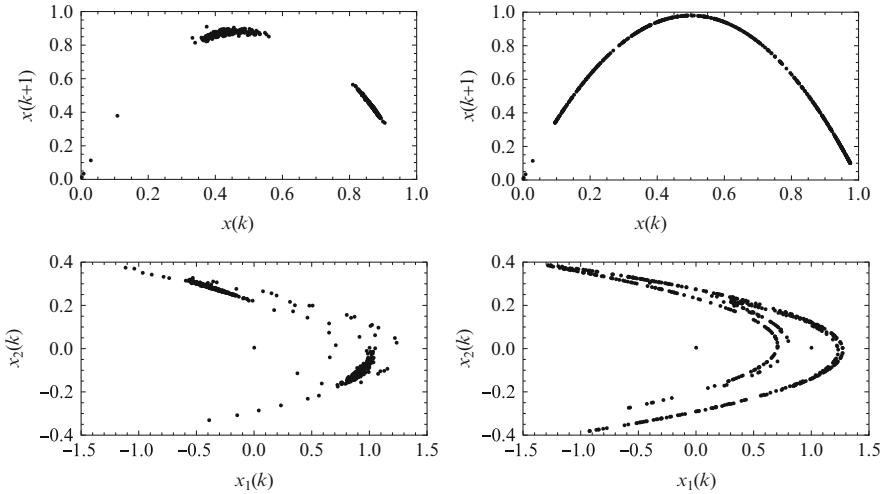

Fig. 5.11 Phase portraits of the average dynamics for a network of $N = 100$ identical maps coupled via a NC network with $n = 5$. The *top row* reports logistic maps with $\varepsilon = 0.05$ (*left*) and $\varepsilon = 0.2$ (*right*). The *bottom row* reports canonical Henon maps with $\varepsilon = 0.05$ (*left*) and $\varepsilon = 0.2$ (*right*). Simulation data use 500 samples

observations from Fig. 5.9, for both the logistic and Henon maps, the norm of the error rapidly approaches zero as the individual systems synchronize when $\varepsilon = 0.2$, while the synchronization error does not approach zero for $\varepsilon = 0.05$ and $\varepsilon = 0.33$. In particular, the error fails to go to zero but stays bounded for $\varepsilon = 0.05$ and it grows unbounded for $\varepsilon = 0.33$ after approximately 20 time steps.

We report the phase portraits for the average dynamics defined by $\frac{1}{N}\sum_{i=1}^{N} x_i(k)$ for $\varepsilon = 0.05$ and $\varepsilon = 0.2$ in Fig. 5.11. Consistently with the previous observations on the synchronization error, the average dynamics of synchronized oscillators corresponds to the attractor of the respective chaotic map, as seen in the right

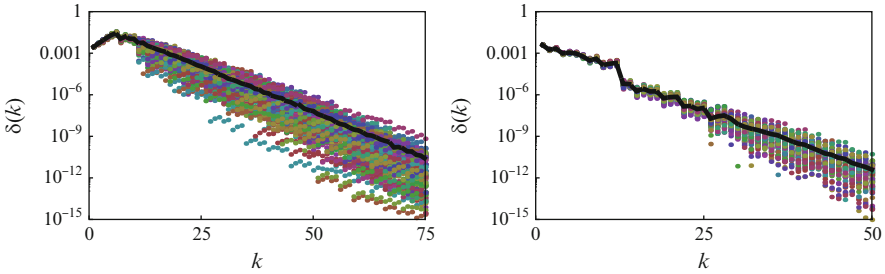

Fig. 5.12 Time evolutions (*markers*) and mean value (*solid line*) of the synchronization error for 75 Monte Carlo simulations with $N = 100$ logistic maps (*left*) and canonical Henon maps (*right*) coupled via a NC network with $n = 5$ and $\varepsilon = 0.2$

panels of Fig. 5.11. Even though the oscillators do not synchronize for $\varepsilon = 0.05$, the coupling is so weak that the average dynamics seem to be confined in the basin of attraction of the respective chaotic map in each case, as seen in the left panels of Fig. 5.11. Conversely, the stronger coupling $\varepsilon = 0.33$ causes the trajectories to grow unbounded after a small number of time steps; thus, the phase portraits for the average logistic and Henon maps are sparse and are not reported here.

To numerically validate these results, Fig. 5.12 presents $\delta(k)$ computed for 75 different realizations of the coupled logistic maps and of the coupled Henon maps with $\varepsilon = 0.2$ and identical initial conditions with components uniformly selected in $[0, 0.001]$. The markers indicate different realizations and the average of the numerical simulations are given by the solid lines. We compute an exponential rate of decay by linear regression of the average error, which equals 0.303 averaging over $k \in [10, 75]$ for the logistic maps and equals 0.412 averaging over $k \in [10, 50]$ for the Henon maps. The theoretical values from the left hand side of (5.32) are 0.690 for the logistic map and 0.823 for the Henon map, which are approximately equal to twice the numerical value found in each simulation case. This finding is consistent with the variational equation (5.27), wherein the trace of $\tilde{\Xi}(k)$ is equal to the expectation of the square of the norm of $\xi(k)$, which corresponds to the expectation of $\delta(k)^2$ for proximal trajectories.

Acknowledgements The authors would like to gratefully acknowledge support from the National Science Foundation under CAREER Grant # CMMI-0745753 and GK-12 Fellows Grant # DGE-0741714.

References

1. Abaid, N., Porfiri, M.: Fish in a ring: spatiotemporal pattern formation in one-dimensional animal groups. J. R. Soc. Interf. **7**(51), 1441–1453 (2010)
2. Abaid, N., Porfiri, M.: Topological analysis of numerosity-constrained social networks. In: Proceedings of the ASME DSCC – Dynamic Systems and Control Conference, p. WeBT3.1 (2010)

3. Abaid, N., Porfiri, M.: Consensus over numerosity-constrained random networks. IEEE Trans. Automat. Contr. **56**(4), 649–654 (2011)
4. Amritkar, R., Hu, C.: Synchronized state of coupled dynamics on time-varying networks. Chaos **16**(1), 015,117 (2006)
5. Arenas, A., Diaz-Guilera, A., Kurths, J., Moreno, Y., Changsong, Z.: Synchronization in complex networks. Phys. Rep. **469**(3), 93–153 (2008)
6. Balanis, C.A.: Antenna Theory: Analysis and Design, 2nd edn. Wiley, New York (1997)
7. Ballerini, M., Cabibbo, N., Candelier, R., Cavagna, A., Cisbani, E., Giardina, I., Lecomte, V., Orlandi, A., Parisi, G., Procaccini, A., Viale, M., Zdravkovic, V.: Interaction ruling animal collective behavior depends on topological rather than metric distance: Evidence from a field study. Proc. Nat. Acad. Sci. **105**(4), 1232–1237 (2008)
8. Barabasi, A.L., Albert, R.: Emergence of scaling in random networks. Science **286**(5439), 509–512 (1999)
9. Belykh, V., Belykh, I., Hasler, M.: Connection graph stability method for synchronized coupled chaotic systems. Physica D **195**(1–2), 159–187 (2004)
10. Bernstein, D.: Matrix Mathematics. Princeton University Press, Princeton, NJ (2005)
11. Bertsekas, D.P., Tsitsiklis, J.N.: Parallel and Distributed Computation: Numerical Methods. Athena Scientific, Belmont, MA (1997)
12. Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., Hwang, D.U.: Complex networks: structure and dynamics. Phys. Rep. **424**(4–5), 175–308 (2006)
13. Buck, J., Buck, E.: Synchronous fireflies. Sci. Am. **234**(5), 74 (1976)
14. Bunke, H., Dickinson, P.J., Kraetzl, M., Wallis, W.D.: A Graph-Theoretic Approach to Enterprise Network Dynamics, chap. Distances, clustering, and small worlds, pp. 147–164. Birkhuser, Boston, MA (2007)
15. Buono, P.L., M., G.: Models of central pattern generators for quadruped locomotion: I. primary gaits. J. Math. Biol. **42**(4), 291–326 (2001)
16. Chen, G., Yu, X. (eds.): Chaos Control Theory and Applications, Lecture Notes in Control and Information Sciences, vol. 292. Springer, New York (2003)
17. Cuomo, K.M., Oppenheim, V.A., Strogatz, S.H.: Synchronization of Lorentz-based chaotic circuits with application to communications. IEEE Trans. Cir. Syst. II **40**(10), 626–633 (1993)
18. DeLellis, P., di Bernardo, M., Gorochowski, T.E., Russo, G.: Synchronization and control of complex networks via contraction, adaptation and evolution. IEEE Circ. Syst. Mag. **10**(3), 64–82 (2010)
19. Duane, G.S., Webster, P.J., Weiss, J.B.: Go-occurrence of northern and southern hemisphere blocks as partially synchronized chaos. J. Atmos. Sci. **56**(24), 4183–4205 (1999)
20. Durrett, R.: Random Graph Dynamics. Cambridge University Press, Cambridge (2007)
21. Erdős, P., Rényi, A.: On the evolution of random graphs. Publications of the Mathematics Institute, Hungarian Academy of Sciences, vol. 5, pp. 17–61 (1960)
22. Erdos P., Renyi, A.: On random graphs. Publicationes Mathematicae **6**, 290–297 (1959)
23. Fagiolo, G.: Clustering in complex directed networks. Phys. Rev. E **76**(2), 026,107 (2007)
24. Fang, Y., Loparo, K.A.: Stochastic stability of jump linear systems. IEEE Trans. Autom. Contr. **47**(7), 1204–1208 (2002)
25. Feng, X., Loparo, K.A., Ji, Y., Chizeck, H.J.: Stochastic stability properties of jump linear systems. IEEE Trans. Autom. Contr. **37**(1), 38–53 (1992)
26. Gleiss, P.M.: Short cycles: Minimum cycle bases of graphs from chemistry and biochemistry. Ph.D. thesis, Universitat Wien (2001)
27. Godsil, C., Royle, G.: Algebraic Graph Theory. Springer, New York (2004)
28. Golub, G.G., Van Loan, C.F.: Matrix Computations, 3rd edn. Johns Hopkins, Baltimore, MD (1996)
29. Gonzalez-Miranda, J.M.: Synchronization and Control of Chaos. Imperial College Press, London (2004)
30. Guevara, M.R., Shirier, A., Glass, L.: Phase-locked rhythms in periodically stimulated heart cell aggregates. Am. J. Physiol. **254**(1), H1–H10 (1988)

31. Hatano, Y., Mesbahi, M.: Agreement over random networks. IEEE Trans Autom. Contr. **50** (11), 1867–1872 (2005)
32. He, J.H., Liu, H.M., Pan, N.: Variational model for ionomeric polymer-metal composite. Polymer **44**, 8195–8199 (2003)
33. Hovareshti, P., Baras, J.S., Gupta, V.: Probabilistic switching and convergence rate in consensus problems. In: Proceedings of the Allerton Conference on Communication, Control, and Computing. Urbana-Champaign, IL (2007)
34. Huang, M., Manton, J.H.: Stochastic consensus seeking with noisy and directed inter-agent communication: Fixed and randomly varying topologies. IEEE Trans. Autom. Contr. **55**(1), 235–241 (2010)
35. Ito, J., Kaneko, K.: Spontaneous structure formation in a network of chaotic units with variable connection strengths. Phys. Rev. Lett. **88**(2), 028,701 (2002)
36. Jadbabaie, A., Lin, J., Morse, A.S.: Coordination of groups of mobile autonomous agents using nearest neighbor rules. IEEE Trans. Autom. Contr. **48**(6), 988–1001 (2003)
37. Ji, Y., Chizeck, H.J., Feng, X., Loparo, K.A.: Stability and control of discrete-time jump linear systems. Contr. Theory Adv. Technol. **7**(2), 247–270 (1991)
38. Juang, J., Liang, Y.H.: Synchronous chaos in coupled map lattices with general connectivity topology. SIAM J. Appl. Dyn. Syst. **7**(3), 755–765 (2008)
39. Kar, S., Moura, J.M.F.: Sensor networks with random links: topology design for distributed consensus. IEEE Trans. Sign. Process. **56**(7), 3315–3326 (2008)
40. Kuramoto, Y.: Chemical Oscillations, Waves, and Turbulence. Springer, New York (1984)
41. Kushner, H.: Introduction to Stochastic Control. Holt, Rinehart, and Winston, New York (1971)
42. Lu, J., Chen, G.: A time-varying complex dynamical network model and its controlled synchronization criteria. IEEE Trans. Autom. Contr. **50**(6), 841–846 (2005)
43. Lu, J., Yu, X., Chen, G.: Chaos synchronization of general complex dynamical networks. Physica A **334**(1–2), 281–302 (2004)
44. Lu, W., Atay, F.M., Jost, J.: Synchronization of discrete-time dynamical networks with time-varying couplings. SIAM J. Math. Anal. **39**(4), 1231–1259 (2007)
45. Ma, J., Song, W.G., Zhang, J., Lo, S.M., Liao, G.X.: k-Nearest-Neighbor interaction induced self-organized pedestrian counter flow. Physica A **389**(10), 2101–2117 (2010)
46. Newman, M.E.J.: Assortative mixing in networks. Phys. Rev. Lett. **89**(20), 208,701 (2002)
47. Newman, M.E.J.: Mixing patterns in networks. Phys. Rev. E **67**(2), 026,126 (2003)
48. Nishikawa, T., Motter, A.: Synchronization is optimal in nondiagonalizable networks. Phys. Rev. E **73**(6), 065,106(R) (2006)
49. Partridge, B.L.: The structure and function of fish schools. Sci. Am. **246**(6), 114–123 (1982)
50. Patterson, S., Bamieh, B., Abbadi, A.E.: Convergence rates of distributed average consensus with stochastic link failures. IEEE Trans. Autom. Contr. **55**(4), 880892 (2010)
51. Pecora, L.M., Carroll, T.L.: Master stability functions for synchronized coupled systems. Phys. Rev. Lett. **80**(10), 2109–2112 (1998)
52. Pereira, S.S., Pages-Zamora, A.: Mean square convergence of consensus algorithms in random WSNs. IEEE Trans. Signal Process. **58**(5), 2866–2874 (2010)
53. Pikovsky, A., Rosemblum, M., Kurths, J.: Synchronization, A Universal Concept in Nonlinear Sciences. Cambridge University Press, Cambridge (2001)
54. Porfiri, M.: A master stability function for stochastically coupled chaotic maps, Europhysics Letters 96, 40014(P1-6), (2011)
55. Porfiri, M., Bollt, E.M., Stilwell, D.J.: Decline of minorities in stubborn societies. Eur. Phys. J. B **57**(4), 481–486 (2007)
56. Porfiri, M., Fiorilli, F.: Global pulse synchronization of chaotic oscillators through fast-switching: theory and experiments. Chaos Solitons Fractals **41**(1), 245–262 (2009)
57. Porfiri, M., Fiorilli, F.: Node-to-node pinning-control of complex networks. Chaos: An Interdiscipl. J. Nonlinear Sci. **19**(1), 013,122 (2009)

58. Porfiri, M., Fiorilli, F.: Experiments on node-to-node pinning control of chua's circuits. Physica D **239**(8), 454–464 (2010)
59. Porfiri, M., Pigliacampo, R.: Master-slave global stochastic synchronization of chaotic oscillators. SIAM J. Appl. Dyn. Syst. **7**(3), 825–842 (2008)
60. Porfiri, M., Stilwell, D.J.: Consensus seeking over random weighted directed graphs. IEEE Trans. Autom. Contr. **52**(9), 1767–1773 (2007)
61. Porfiri, M., Stilwell, D.J., Bollt, E.M.: Synchronization in random weighted directed networks. IEEE Trans. Cir. Syst. I **55**(10), 3170–3177 (2008)
62. Porfiri, M., Stilwell, D.J., Bollt, E.M., Skufca, J.D.: Random talk: Random walk and synchronizability in a moving neighborhood network. Physica D **224**(1-2), 102–113 (2006)
63. Rangarajan, G., Ding, M.: Stability of synchronized chaos in coupled dynamical systems. Phys. Lett. A **296**(4–5), 204–209 (2002)
64. Ren, W., Beard, R.W.: Distributed Consensus in Multi-vehicle Cooperative Control. Springer, Berlin (2008)
65. Stefanski, A., Wojewoda, J., Kapitaniak, T., Yanchuk, S.: Simple estimation of synchronization threshold in ensembles of diffusively coupled chaotic systems. Phys. Rev. E **70**(2), 026,217 (2004)
66. Stilwell, D.J., Bollt, E.M., Roberson, D.G.: Sufficient conditions for fast switching synchronization in time varying network topologies. SIAM J. Appl. Dyn. Syst. **5**(1), 140–156 (2006)
67. Sumpter, D.J.T.: Collective Animal Behavior. Princeton University Press, Princeton (2009)
68. Sun, J., Bollt, E.M., Nishikawa, T.: Complex Sciences Part 1, chap. Synchronization stability of coupled near-identical oscillator network, pp. 900–911. Springer, Berlin (2009)
69. Tahbaz-Salehi, A., Jadbabaie, A.: A necessary and sufficient condition for consensus over random networks. IEEE Trans. Automat. Contr. **53**(3), 791–795 (2008)
70. Tegeder, R.W., Krause, J.: Density dependence and numerosity in fright stimulated aggregation behaviour of shoaling fish. Phil. Trans. R. Soc. B **350**(1334), 381–390 (1995)
71. Watts, D.J., Strogatz, S.H.: Collective dynamics of 'small-world' networks. Nature **393**, 440–442 (1998)
72. Womelsdorf, T., Fries, P.: The role of neuronal synchronization in selective attention. Curr. Opin. Neurobiol. **17**(2), 154–160 (2007)
73. Wu, C.W.: Synchronization and convergence of linear dynamics in random directed networks. IEEE Trans. Automat. Contr. **51**(7), 1207–1210 (2006)
74. Wu, C.W.: Synchronization in Complex Networks of Nonlinear Dynamical Systems. World Scientific, New York (2007)
75. Yanchuk, S., Stefanski, A., Kapitaniak, T., Wojewoda, J.: Dynamics of an array of mutually coupled semiconductor lasers. Phys. Rev. E **73**(1), 016,209 (2006)
76. Zhao, J., Hill, D.J., Liu, T.: Synchronization of complex dynamical networks with switching topology: A switched system point of view. Automatica **45**(11), 2502–2511 (2009)
77. Zhou, J., Wang, Q.: Convergence speed in distributed consensus over dynamically switching random networks. Automatica **45**(6), 1455–1461 (2009)

Part 4
Nonlinear Dynamics in Transport and Mechanical Engineering

Chapter 6
Nonlinear Dynamics of a Road Vehicle Running into a Curve

Gianpiero Mastinu, Fabio Della Rossa, and Carlo Piccardi

6.1 Introduction

In order to describe the stability of an automobile running either straight or into a curve, a linear mathematical model could be used to take into account the vast majority of actual situations occurring in everyday practice. There are however rare but important cases in which the dynamics of automobiles need to be described by non linear models. An important example is given by accident avoidance manoeuvres, in which a well designed vehicle can make the difference between a dramatic or an even unnoticed event. The aim of this chapter is to focus on the description of the non linear dynamic behavior of automobiles running either straight ahead or into bends on an even surface. The non linearities of the automobile model refer to the non linear tyre characteristics.

A general approach for studying and classifying the behaviors of a nonlinear system when one or a few parameters are varied is *bifurcation analysis* [11]. In the recent literature on road vehicle dynamics [4, 9], bifurcation analysis has been successfully applied to study driveline vibrations [1], vertical vibrations [17, 20], tractor semi-trailer vibrations [1, 16], and the interaction among vehicles in the traffic (vehicle following and overtaking) [15]. Nonlinear controls for enhancing stability have also been conceived based on bifurcation analysis [6, 12].

In this chapter we concentrate on the basic issue of the global stability of an automobile, by deriving non-dimensional equations of motion of a car running either straight or into a curve. By means of non dimensional analysis, the minimum set of parameters describing the non linear behavior of the automobile is derived.

G. Mastinu (✉)
Politecnico di Milano, Dipartimento di Meccanica, Milano, Italy
e-mail: gianpiero.mastinu@polimi.it

F.D. Rossa • C. Piccardi
Politecnico di Milano, Dipartimento di Elettronica e Informazione, Milano, Italy
e-mail: fabio.dellarossa@mail.polimi.it; carlo.piccardi@polimi.it

S. Banerjee et al. (eds.), *Applications of Chaos and Nonlinear Dynamics in Science and Engineering - Vol. 2*, Understanding Complex Systems,
DOI 10.1007/978-3-642-29329-0_6,

A number of key bifurcation diagrams are produced to highlight the most significant non linear behaviors.

The chapter is organized as follows. First, we present the model used for the analysis. The derivation is rather detailed in order to allow the reader who is not a specialist on vehicle dynamics to understand the whole topic. Then a non dimensional analysis is performed. Bifurcations of equilibria are computed for a number of different combinations of front and rear tyre characteristics. The cases of particular interest are highlighted.

6.2 Mechanical Model and Mathematical Model

In this section a mechanical model of a vehicle running on a flat and even surface is first shown, and the related mathematical model is then derived. For simplicity, the mechanical model of the vehicle is a flat rigid body with four wheels. Some peculiar subsystems of the vehicle (such as driveline, suspension system, etc.) are not introduced in order to keep the model complexity at a level that allows a manageable bifurcation analysis. Despite the study presented hereafter is based on a simplified vehicle model, the results are significant and important from an engineering point of view. As a matter of fact, such a simple model can reproduce fairly well the actual vehicle motion in cases in which safety is the main issue, for example when the vehicle is running on slippery ground surfaces (i.e., covered with ice or snow). In this case, since the forces acting on the suspension system are small, their influence on the motion of the vehicle is almost negligible so that the suspensions can be removed from the model [14]. All the control systems for enhancing stability (such as ABS and ESP) are often designed by means of the simple model presented in this chapter, the validation tests being performed on slippery surfaces [2]. The derivation presented hereafter refers to [16]. The reader who is not an expert in vehicle dynamics can follow step by step the derivation of the ordinary differential equations which describe the motion of the vehicle and which will be used in the bifurcation analysis.

6.2.1 Reference System

The reference system adopted to write the equations of motion and to describe the fundamental kinematics is depicted in Fig. 6.1.

The x-axis is directed towards the longitudinal axis of the vehicle body, whereas the y-axis is orthogonal to the previous one and passes through the centre of gravity G. The vertical z-axis (not drawn in the planar figure) passes through G too. The three reference axes are defined by the respective vectors $\mathbf{i}, \mathbf{j}, \mathbf{k}$. The velocity at the centre of gravity G is $\mathbf{v}_G$ and its components along x and y are denoted as u e v. The centre of gravity G lies at the ground level. Since the motion takes place exclusively in the $x - y$ plane, the vertical component z vanishes. The angle between $\mathbf{v}_G$ and x is β (*yaw angle*). Obviously $\tan(\beta) = v/u$. The angular velocity around the z-axis, is the *yaw*

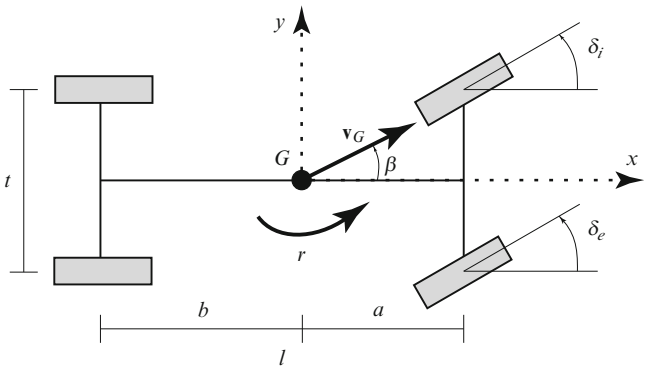

Fig. 6.1 Reference system

angular velocity $\boldsymbol{\Omega} = r\mathbf{k}$. The (horizontal) distance between the centre of the fore axle and the centre of the rear axle is the wheelbase l. The respective distances a and b of G from the centre of the fore and of the rear axle define the position of the centre of gravity, i.e., $l = a + b$. The distance between the centres of the two wheels of the same axle is the track t. For simplicity, the front and rear tracks are supposed to be equal (this is almost true for most vehicles). The mass of the vehicle is m, and $\bar{I}$ is the inertia tensor of the vehicle. We assume that $\bar{I}$ is of the form

$$\bar{I} = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix},$$

which implies that the principal inertia axes are coincident with the axes of the reference system in Fig. 6.1. The error introduced by this simplification is negligible for the presented analyzes.

6.2.2 Steering Angles

The steering angle of a wheel is the angle between a line directed as the meridian plane of the wheel and the longitudinal axis of the vehicle. To preserve the correct kinematics (at least at vanishing speed), the two steering wheels have different steering angles. The inner wheel is steered by the angle δ_i, which is larger than the one of the outer wheel δ_e (see Fig. 6.2).

From Fig. 6.2 we obtain

$$\tan(\delta_e) = \frac{l}{\rho + t/2},$$
$$\tan(\delta_i) = \frac{l}{\rho - t/2}.$$

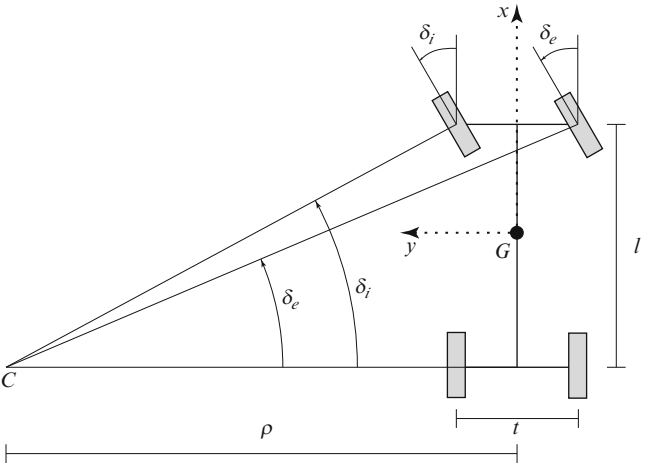

Fig. 6.2 Different steering angles for the outer and inner wheel

Eliminating ρ from the above equations, we obtain

$$\frac{1}{\tan(\delta_e)} = \frac{t}{l} + \frac{1}{\tan(\delta_i)},$$

which, expanded in Taylor series, gives

$$\delta_e = \delta_i - \frac{t}{l}\delta_i^2 + O(\delta_i^3).$$

For practical values of the steering angles (not larger than a few degrees) we can fairly assume that the steering angles of the outer and inner wheels are equal, i.e., $\delta = \delta_e \approx \delta_i$.

6.2.3 Computation of the Acceleration at the Centre of Gravity

Let us compute the acceleration at the centre of gravity, that will be useful to derive the equations of motion. The linear and angular speed components at G are given by

$$\mathbf{v}_G = u\mathbf{i} + v\mathbf{j},$$
$$\Omega = r\mathbf{k}$$

Since

$$\frac{d\mathbf{i}}{dt} = r\mathbf{j},$$
$$\frac{d\mathbf{j}}{dt} = -r\mathbf{i},$$

we obtain

$$\begin{aligned}\mathbf{a}_G &= \frac{d\mathbf{v}_G}{dt} = \frac{du}{dt}\mathbf{i} + u\frac{d\mathbf{i}}{dt} + \frac{dv}{dt}\mathbf{j} + v\frac{d\mathbf{j}}{dt} = \\ &= \frac{du}{dt}\mathbf{i} + ur\mathbf{j} + \frac{dv}{dt}\mathbf{j} - vr\mathbf{i} = \\ &= \left(\frac{du}{dt} - vr\right)\mathbf{i} + \left(\frac{dv}{dt} + ur\right)\mathbf{j} = \\ &= a_x\mathbf{i} + a_y\mathbf{j}\end{aligned}$$

The components of the acceleration at G are then given by

$$a_x = \frac{du}{dt} - vr,$$
$$a_y = \frac{dv}{dt} + ur = \frac{dv}{dt} + \frac{u^2}{\rho},$$

where ρ is the radius of the path followed by G ($u = r\rho$). At steady-state the derivatives vanish and we have

$$\tilde{a}_x = -vr, \qquad \tilde{a}_y = \frac{u^2}{\rho}.$$

In the following, we will use the standard notation $du/dt = \dot{u}$, $dv/dt = \dot{v}$.

6.2.4 Equations of Motion

We consider the motion of the vehicle (a flat rigid body) in the $x - y$ plane. We have three state variables, namely u, v, and r, and we have to write three equations describing the dynamic equilibrium, namely the two equations describing the balance of the forces acting along the x-axis and y-axis, respectively, and the equation describing the balance of the moments of the forces around the z-axis. The forces acting on the vehicle model are depicted in Fig. 6.3.

The three balance equations read

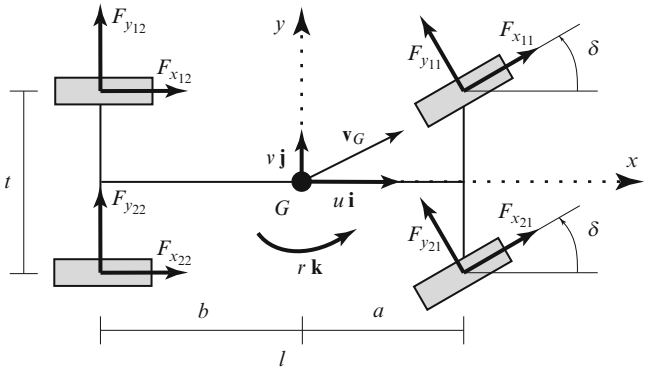

Fig. 6.3 Forces acting on the vehicle model

$$
\begin{aligned}
m(\dot{u} - vr) &= (F_{x_{11}} + F_{x_{12}})\cos(\delta) - (F_{y_{11}} + F_{y_{12}})\sin(\delta) + (F_{x_{21}} + F_{x_{22}}) - \frac{1}{2}\rho S C_x u^2,\\
m(\dot{v} + ur) &= (F_{x_{11}} + F_{x_{12}})\sin(\delta) + (F_{y_{11}} + F_{y_{12}})\cos(\delta) + (F_{y_{21}} + F_{y_{22}}),\\
I_z\dot{r} &= a[(F_{x_{11}} + F_{x_{12}})\sin(\delta) + (F_{y_{11}} + F_{y_{12}})\cos(\delta)] - b(F_{y_{21}} + F_{y_{22}}) +\\
&\quad - \frac{t}{2}\left[(F_{x_{11}} - F_{x_{12}})\cos(\delta) + (F_{x_{21}} - F_{x_{22}})\cos(\delta) - (F_{y_{11}} - F_{y_{12}})\sin(\delta)\right].
\end{aligned}
$$

The above equations can be simplified by assuming that the force at the left side of an axle is equal to the corresponding one at the right side. Additionally we will add the forces at the left and right side of an axle, obtaining

$$
\begin{aligned}
F_{x_1} &= F_{x_{11}} + F_{x_{12}}, \qquad & F_{x_2} &= F_{x_{21}} + F_{x_{22}},\\
F_{y_1} &= F_{y_{11}} + F_{y_{12}}, \qquad & F_{y_2} &= F_{y_{21}} + F_{y_{22}},
\end{aligned}
$$

Moreover, assuming that the steering angle δ is small (typical values are between -15 and $+15$ degrees) we can set $\cos(\delta) \approx 1$ e $\sin(\delta) \approx \delta$. Thus

$$
\begin{aligned}
m(\dot{u} - vr) &= F_{x_1} + F_{x2} - F_{y_1}\delta - \frac{1}{2}\rho S C_x u^2,\\
m(\dot{v} + ur) &= F_{x_1}\delta + F_{y_1} + F_{y_2},\\
I_z\dot{r} &= F_{x_1}\delta a + F_{y_1} a - F_{y_2} b
\end{aligned}
$$

By means of the above described approximations we have obtained the so called *single-track model*, in which the vehicle is actually reduced to a rigid pole with two tires. It is worth noticing that the same equations can actually be obtained by starting directly from this assumption, namely by using the mechanical model of Fig. 6.4.

The angles α_1 and α_2 in Fig. 6.4 are the *lateral slip angles* and play a fundamental role in the vehicle dynamics, as it will be detailed in Sect. 6.2.5. The centre of

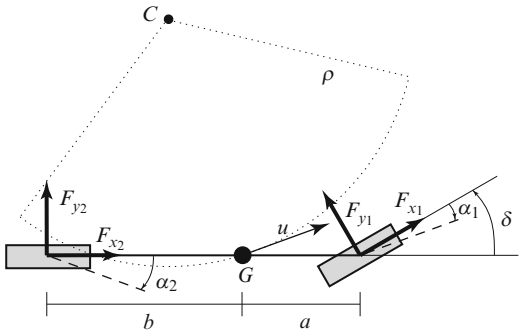

Fig. 6.4 Representation of the *single-track model*

rotation of the vehicle model is denoted by C in Fig. 6.4. In the same figure the radius ρ of the trajectory of G is reported.

In order to obtain the simplest possible set of equations for the subsequent bifurcation analysis, we now introduce the following additional assumptions:

- The forward speed u is constant.
- The drag forces (rolling resistances and aerodynamic resistances) are vanishing.
- No longitudinal forces are acting at the wheels.

The first assumption implies that the first balance equation actually reduces to an algebraic one, so that the mathematical model of the vehicle has only two differential equations, i.e., it is a two-degree-of-freedom mechanical model. The aerodynamic forces are disregarded because their effects can be assumed to be negligible (see [16]). The same holds for the rolling resistance. If we consider a rear wheel drive vehicle model, we have $F_{x1} = 0$, which allows to cancel the term in the second balance equation. Since $F_{x1} = 0$ is related to the vanishing drag forces, such a term can be canceled also for a front wheel drive vehicle model.

Thus the differential equations describing the motion of an automobile running on an even surface are

$$\begin{aligned} m(\dot{v} + ur) &= F_{y_1} + F_{y_2}, \\ I_z \dot{r} &= F_{y_1} a - F_{y_2} b. \end{aligned} \tag{6.1}$$

The non linearities of such a second order autonomous system can be found in the expressions of the forces F_{y1} and F_{y2}, which depend on the lateral slips α_1 and α_2, which in turn can be expressed as function of the state variables.

6.2.5 *Lateral Slip as a Function of the State Variables*

The lateral slip α of a wheel is the angle between the line directed as the meridian plane of the wheel and the velocity vector at contact point of the wheel on the

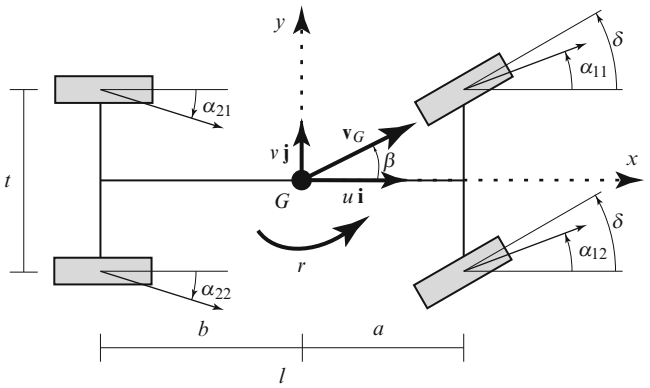

Fig. 6.5 Lateral slips

ground. In general a vehicle model like the one depicted in Fig. 6.1 has four different lateral slips, one per each wheel (see Fig. 6.5).

We have assumed that the vehicle can be described by one single rigid body. Thus the speed of the contact point on the ground of each wheel can be derived as a function of the speed of the centre of gravity $\mathbf{v}_G$ and of the yaw velocity r. For a conventional two wheel steer vehicle, the following relationships hold:

- Front left wheel:

$$\mathbf{v}_{11} = (u - \frac{1}{2}rt, v + ra) \quad \Longrightarrow \quad \tan(\delta - \alpha_{11}) = \frac{v + ra}{u - \frac{1}{2}rt}$$

- Front right wheel:

$$\mathbf{v}_{12} = (u + \frac{1}{2}rt, v + ra) \quad \Longrightarrow \quad \tan(\delta - \alpha_{12}) = \frac{v + ra}{u + \frac{1}{2}rt}$$

- Rear left wheel:

$$\mathbf{v}_{21} = (u - \frac{1}{2}rt, v - rb) \quad \Longrightarrow \quad \tan(-\alpha_{12}) = \frac{v - rb}{u - \frac{1}{2}rt}$$

- Rear right wheel:

$$\mathbf{v}_{22} = (u + \frac{1}{2}rt, v - rb) \quad \Longrightarrow \quad \tan(-\alpha_{22}) = \frac{v - rb}{u + \frac{1}{2}rt}$$

In general, the radius of the path is much bigger than the track ($\rho \gg t$) so that, recalling that $u = |r|\, R$, we have $u \gg |r|t/2$. Thus

$$\tan(\delta - \alpha_{11}) \approx \tan(\delta - \alpha_{12}) \approx \frac{v + ra}{u},$$
$$\tan(-\alpha_{21}) \approx \tan(-\alpha_{22}) \approx \frac{v - rb}{u}.$$

and therefore

$$\alpha_1 = \alpha_{11} \approx \alpha_{12},$$
$$\alpha_2 = \alpha_{21} \approx \alpha_{22}$$

In other words, we can assume that the fore wheels have the same lateral slip α_1 and the rear wheels have the same lateral slip α_2. Moreover, the state variables v ed r and the lateral slips α_1 e α_2 are related by

$$\tan(\delta - \alpha_1) = \frac{v + ra}{u},$$
$$\tan(-\alpha_2) = \frac{v - rb}{u}$$

Noticing that $u \gg |v|$, $u \gg |v + ra|$, and $u \gg |v - rb|$, we obtain

$$\delta - \alpha_1 = \frac{v + ra}{u},$$
$$-\alpha_2 = \frac{v - rb}{u}. \tag{6.2}$$

The above equations establish a one-to-one (linear) relationship between the state variables v, r and the lateral slips α_1, α_2. This implies that the latter can be used as state variables as well.

6.2.6 *Equations of Motion as Functions of the Lateral Slips*

From (6.1) and (6.2), it is possible to rewrite the equations of motion as a second-order ordinary differential equations (ODE) of the form $\dot{\alpha} = f(\alpha, u, \delta)$. In detail:

$$\dot{\alpha}_1 = \frac{g}{u}\left[(\delta - \alpha_1 + \alpha_2)\frac{u^2}{gl} - \frac{F_{y_1} + F_{y_2}}{mg} - \frac{a(F_{y_1}a - F_{y_2}b)}{I_z g}\right],$$
$$\dot{\alpha}_2 = \frac{g}{u}\left[(\delta - \alpha_1 + \alpha_2)\frac{u^2}{gl} - \frac{F_{y_1} + F_{y_2}}{mg} - \frac{b(F_{y_2}b - F_{y_1}a)}{I_z g}\right], \tag{6.3}$$

where g is the gravitational acceleration and $l = a + b$ is the wheelbase. The lateral forces F_{yi} $(i = 1, 2)$ can be expressed as functions of the slip angles α_i by the *tyre characteristics*, which are non linear and take the analytical form [16]:

$$\begin{aligned} F_{y_1}(\alpha_1) &= D_1 \sin(C_1 \tan^{-1}(B_1\alpha_1 - E_1(B_1\alpha_1 - \tan^{-1}(B_1\alpha_1)))), \\ F_{y_2}(\alpha_2) &= D_2 \sin(C_2 \tan^{-1}(B_2\alpha_2 - E_2(B_2\alpha_2 - \tan^{-1}(B_2\alpha_2)))), \end{aligned} \tag{6.4}$$

where

$$D_1 = \mu_1 \frac{mg}{l} b, \qquad D_2 = \mu_2 \frac{mg}{l} a,$$

and μ_1 and μ_2 are the friction coefficients of the front and the rear tyres, respectively.

The parameter set is composed by three subsets: the vehicle parameters (m, I_z, a, l), the tyre parameters $(\mu_i, B_i, C_i, E_i, i = 1, 2)$, and the parameters (u, δ) which represent the driver inputs. We will see that the number of independent parameters can actually be reduced by setting the equations of motion in non dimensional form.

6.2.7 Equations of Motion in Non dimensional Form

The above model can be further simplified by reducing the number of parameters, more precisely by eliminating a few of the parameters characterizing the vehicle. This leads to a "universal model" for the vehicles of this class. We first define the new non dimensional time variable

$$\tau = \frac{u}{l} t,$$

which depends on the ratio between the forward velocity and the wheelbase of the vehicle. Notice that this substitution does not alter the dynamics of the model, since varying the forward velocity u merely means going over the same trajectories with different velocity ("orbital equivalence"). Then, by inserting (6.4) into (6.3), and exploiting the empirical relationship $I_z = mab$, we obtain

$$\begin{aligned} \frac{d\alpha_1}{d\tau} &= \delta - \alpha_1 + \alpha_2 - \frac{\mu_1 lg}{u^2} \sin(f_{y_1}(\alpha_1)) - \frac{2\mu_2 ag}{u^2} \sin(f_{y_2}(\alpha_2)), \\ \frac{d\alpha_2}{d\tau} &= \delta - \alpha_1 + \alpha_2 - \frac{2\mu_1 bg}{u^2} \sin(f_{y_1}(\alpha_1)) - \frac{\mu_2 lg}{u^2} \sin(f_{y_2}(\alpha_2)), \end{aligned}$$

where

$$f_{y_1}(\alpha_1) = C_1 \tan^{-1}(B_1\alpha_1 - E_1(B_1\alpha_1 - \tan^{-1}(B_1\alpha_1))),$$
$$f_{y_2}(\alpha_2) = C_2 \tan^{-1}(B_2\alpha_2 - E_2(B_2\alpha_2 - \tan^{-1}(B_2\alpha_2))).$$

By means of this transformation, we cancel the dependence of the model upon the mass and the inertia of the vehicle. Moreover, the importance of the position of the centre of mass as a crucial design parameter is emphasized. Defining now the new non dimensional parameter

$$\xi = \frac{a}{l} \quad \Rightarrow \quad a = l\xi, \quad b = l(1-\xi),$$

and $\Theta = lg/u^2$, we obtain the non dimensional form of model (6.3) as

$$\frac{d\alpha_1}{d\tau} = \delta - \alpha_1 + \alpha_2 - \Theta\mu_1 \sin(f_{y_1}(\alpha_1)) - 2\xi\Theta\mu_2 \sin(f_{y_2}(\alpha_2)),$$
$$\frac{d\alpha_2}{d\tau} = \delta - \alpha_1 + \alpha_2 - 2(1-\xi)\Theta\mu_1 \sin(f_{y_1}(\alpha_1)) - \Theta\mu_2 \sin(f_{y_2}(\alpha_2)). \tag{6.5}$$

In principle, a bifurcation analysis in the three dimensional parameter space (δ, Θ, ξ) would provide a full description of the different behavior that a generic car has during cornering. Notice that (δ, Θ) are control parameters (the steering angle and a function of the forward velocity), while ξ is a design parameter. In the following, for simplicity we will keep ξ at a fixed value, and we will perform a bifurcation analysis with respect to δ and Θ. To preserve the full physical interpretation of the results, we will always specify the forward vehicle speed u besides the corresponding Θ value.

6.3 Bifurcation Analysis

The most general way for studying (6.5) is provided by *bifurcation analysis*, an approach pertaining to non linear systems theory and aiming at analyzing and classifying the behaviors of a system when one or more parameters are varied [11]. Usually, the analysis starts by studying an equilibrium, and by deriving the dependence of its coordinates on a parameter. In doing so, a *bifurcation*, namely a structural change in the system behavior, is possibly detected. A complete catalogue of the system behaviors with respect to the admissible parameters' values could be derived by systematically analyzing all the equilibria and their bifurcations.

An introductory example is presented in Fig. 6.6. The upper-left panel displays the dependence of the equilibria coordinates (α_1, α_2) on the parameter δ. For small values of the steering angle ($\delta < \delta^C$) three equilibria are found, denoted by e_j^i

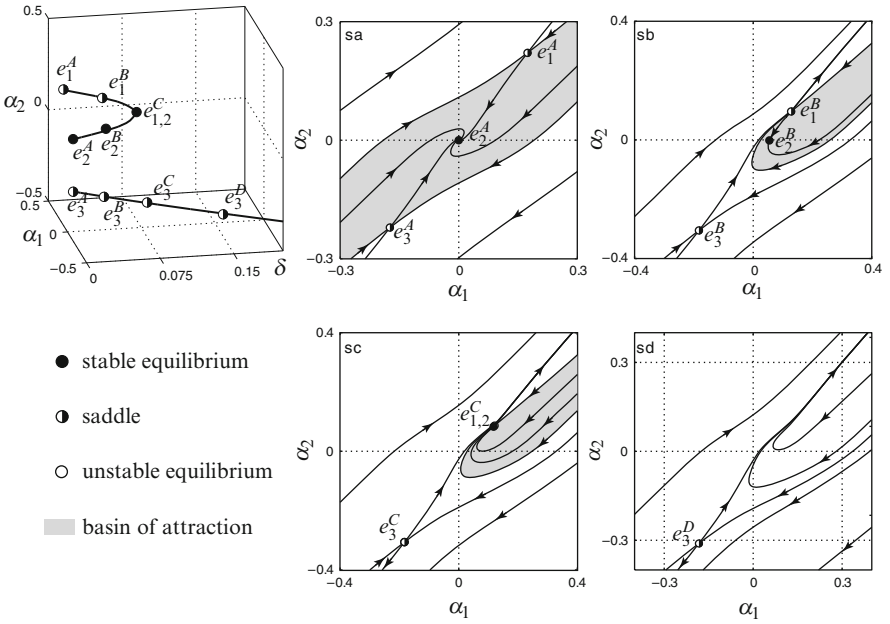

Fig. 6.6 An example of bifurcation analysis of the vehicle model of Fig. 6.4 (6.5) with respect to the parameter δ, with $\Theta = 0.060$ (the speed is fixed at $u = 20, \Theta = 24.1/u^2$), $\xi = 0, 386$. The phase portraits (*right panels*) correspond to the δ values marked from Ⓐ to Ⓓ in the bifurcation diagram (*upper-left panel*). No stable equilibria exist for $\delta > \delta^C$ (phase portrait Ⓓ)

($i = A, B, j = 1, 2, 3$) in the figure. For large steering angle ($\delta > \delta^C$), only one unstable equilibrium ${e_3}^D$ exists. The curves are obtained by means of a *continuation algorithm* [11]. If we increase δ from 0, at $\delta = \delta^C = 0.0755$ the equilibria e_1 and e_2 collide and disappear in a *saddle-node bifurcation*. This corresponds to a qualitative change in the behavior of the non linear system. The saddle-node bifurcation divides the entire region of possible running conditions into two well distinct parts, since only for $\delta < \delta^C$ a stable equilibrium exists.

Even in this basic example there is however one non trivial phenomenon that only a careful bifurcation analysis is able to detect. Compare the phase planes Ⓐ and Ⓑ in Fig. 6.6: an important, qualitative change takes place between them, since the basin of attraction of e_2 (the only stable equilibrium) changes its topology and dramatically shrinks: this greatly restricts the set of perturbations that can safely be absorbed. The phenomenon is due to a *heteroclinic bifurcation* occurring at a value of δ ($\delta = 0.072$) between δ^A and δ^B, a type of bifurcation that can be detected and studied with standard tools.

Since bifurcations play a strategic role for understanding the vehicle dynamics, we will focus on finding the combinations of the input parameters (δ, Θ) at which system (6.5) undergoes a bifurcation. To preserve the full physical interpretation of the results, since $\Theta = \Theta(u)$, the forward speed u will be used for representation. The combinations of parameters $(\delta, \Theta(u))$ define the *bifurcation curves* in the $(\delta, \Theta(u))$

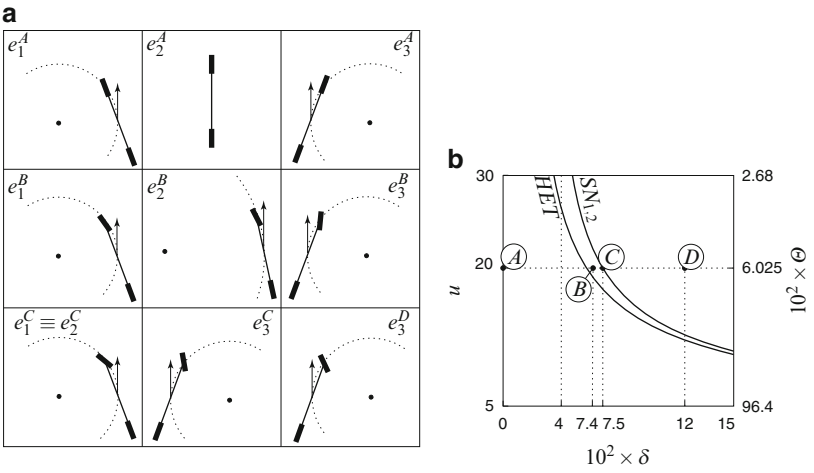

Fig. 6.7 (**a**) Steady-state attitudes of the vehicle of Fig. 6.4 at the equilibria e_j^i ($j = 1, 2, 3$, i = A, B,C,D). (**b**) Bifurcation diagram of the single-track model of Fig.6.4 with respect to the input parameters (δ, Θ) (case UN_1, see Sect. 6.3.1.2). $\xi = 0.386$. u is the forward speed, $\Theta = 24.1/u^2$. $SN_{1,2}$ is a curve of saddle-node bifurcation of equilibria e_1 and e_2, *HET* is a curve of heteroclinic bifurcation of the saddles e_1 and e_3. Points Ⓐ to Ⓓ correspond to the phase planes of Fig. 6.6

-plane, which separate regions with different qualitative behaviors. Consider, for example, the problem of deriving the saddle-node (SN) bifurcation curve. We already have one point of this curve, because the analysis of Fig. 6.6 revealed, for fixed $\Theta = 0.060$ ($u = 20$), a saddle-node bifurcation at $\delta = 0.0755$. By a proper continuation algorithm [11] the entire SN curve in the $(\delta, \Theta(u))$ can be derived. A similar approach can be followed for obtaining the heteroclinic (HET) bifurcation curve. The results are depicted in the right panel of Fig. 6.7. Any combination $(\delta, \Theta(u))$ on the left of the SN curve refers to cases in which a stable equilibrium exist, whereas no stable equilibria are found on the right of SN. Moreover, by crossing the HET curve from the left to the right, the basin of attraction of the stable equilibrium shrinks dramatically and changes topology (Fig. 6.6).

In the remainder of this section we will analyze 7 cases of the vehicle model of (6.5), diversified by different values of the tyre parameters appearing in (6.4). Specifically, we will consider 4 cases referring to understeering vehicles and 3 cases referring to oversteering vehicles. An understeering vehicle naturally increases the radius of the bend when the speed is increased. Viceversa, an oversteering vehicle naturally decreases the radius of the bend when the speed is increased (see [16]). We will discover many different bifurcations, from the sudden disappearance of stable equilibria, to the birth of oscillatory behaviors (limit cycles).

6.3.1 Understeering Cases

6.3.1.1 Case UN_0

$(\xi = 0.386,\ \mu_1 = 0.7,\ \mu_2 = 0.9,\ B_1 = B_2 = -10,\ C_1 = C_2 = 1,\ E_1 = E_2 = 0)$

This case refers to what should be obtained by a proper design of an automobile. Only one stable equilibrium exists for every combination of δ and Θ (i.e., for every combination of steering angle and forward speed u). Since a globally stable equilibrium occurs, any perturbation is absorbed, which implies that the actual vehicle even if not controlled would reach the initial steady-state after *any* disturbance. Thus, such a vehicle can be controlled by the driver in a relatively easy way. This case was discussed in [16].

The other cases analyzed below basically take place as a consequence of a wrong design of the tyre-chassis system. More often the cases analyzed below are due to special occurrences such as unequal wear or irregular pressure of the tyres, or excessive traction/braking forces applied to the tyres (see again [16]).

6.3.1.2 Case UN_1

$(\xi = 0.386,\ \mu_1 = 0.8,\ \mu_2 = 0.9,\ B_1 = -20,\ B_2 = -10,\ C_1 = C_2 = 1,\ E_1 = E_2 = 0)$

This case has already been considered in Fig. 6.6. Three equilibria occur (two unstable and one stable) and the basin of attraction suddenly shrinks as Θ is decreased (i.e., the forward speed u is increased [13]). Two bifurcations take place: the first one is heteroclinic and its occurrence dramatically modifies the basin of attraction. The second bifurcation is a saddle-node one.

6.3.1.3 Case UN_2

$(\xi = 0.386,\ \mu_1 = \mu_2 = 0.9,\ B_1 = -10,\ B_2 = -14,\ C_1 = C_2 = 1.3,\ E_1 = -5,\ E_2 = -8)$

As partially analysed in [18], three bifurcations occur in this case: heteroclinic, homoclinic, and a special case of Bogdanov–Takens bifurcation. We refer to the Appendix for details on the the Bogdanov–Takens bifurcation, although the basic phenomena are described below. The combined action of the homoclinic and Bogdanov–Takens bifurcations gives rise to an unstable limit cycle around the stable equilibrium, which limits its basin of attraction. Let us analyse the phase portraits of Fig. 6.8:

Ⓐ This is the most desirable situation for this case, however the basin of attraction is limited so that only bounded perturbations can be absorbed.

Ⓑ Heteroclinic bifurcation. The unstable manifold of the saddle point e_3 is connected to the other saddle point e_1 instead of being connected to the stable equilibrium, as in the previous case Ⓐ. The stable equilibrium is close to the border of the basin of attraction. This occurrence is very dangerous as the driver

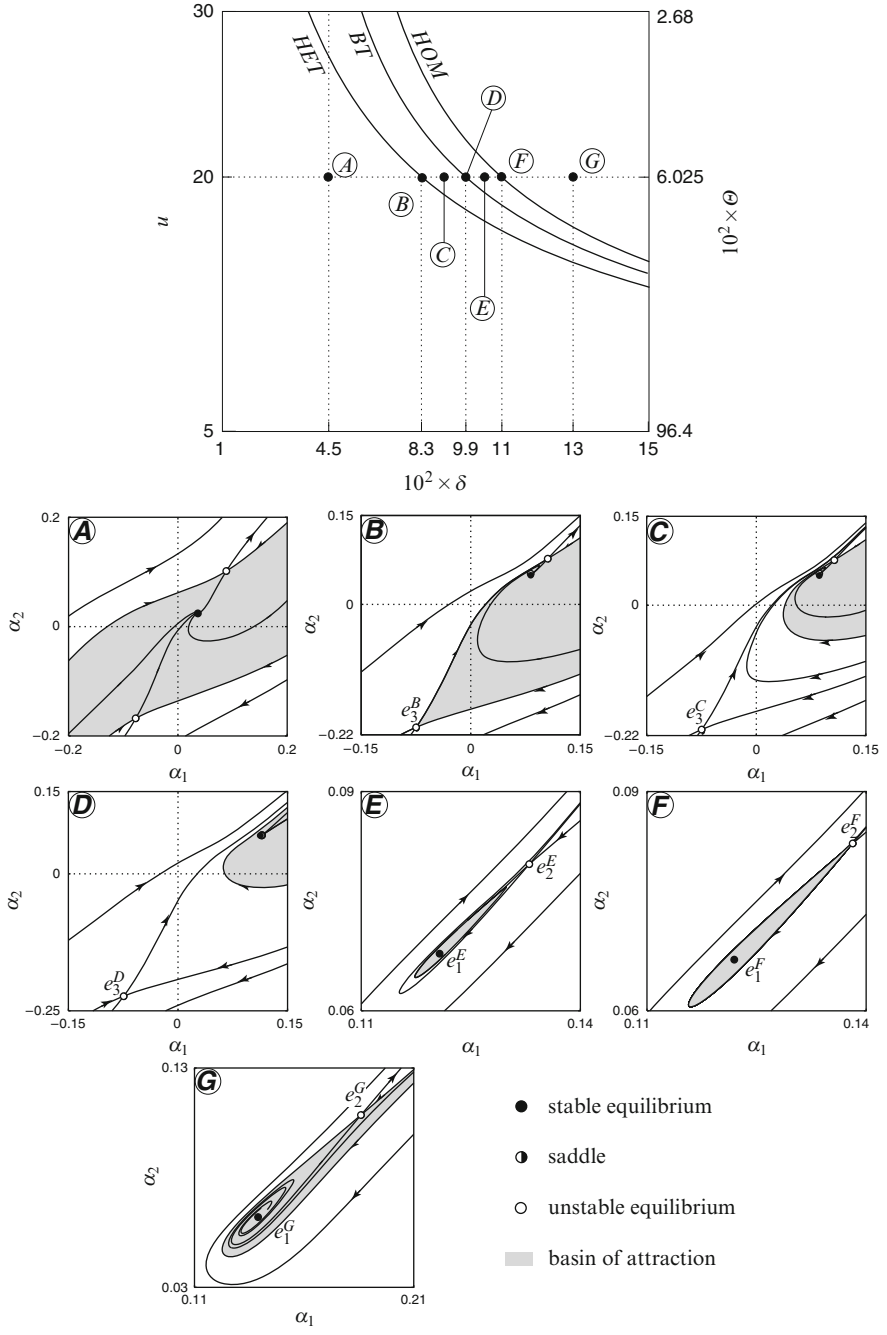

Fig. 6.8 Bifurcation diagram of the single-track model of Fig. 6.4 with respect to the input parameters (δ, Θ) (case UN_2). $\xi = 0.386$. u is the forward speed, $\Theta = 24.1/u^2$. The phase portraits Ⓐ to Ⓖ correspond to the (δ, Θ) values reported in the bifurcation diagram (*central panel*). *BT* is a curve of Bogdanov–Takens bifurcations

could be required to cope with a diverging system after a small perturbation given in the 'wrong' direction.

Ⓒ The basin of attraction is now defined by the stable manifolds of the saddle e_1 only. The basin of attraction is further shrunk. Again the stable equilibrium is almost on the border of its basin of attraction, which is a dangerous occurrence.

Ⓓ Bogdanov–Takens bifurcation. It is a complex bifurcation that occurs in this system because of physical phenomena taking place at the same parameters' values. The equilibria e_1 (saddle) and e_2 (stable focus) merge but do not disappear. This is due to the fact that the maximum values of the forces at the front and rear tyres are the same (this fact can be proved by inserting the tyre parameters into (6.4)). As a matter of fact, due to this occurrence a transcritical bifurcation takes place, instead of a saddle-node one. An additional bifurcation (subcritical Hopf) actually occurs at this point, giving birth to an unstable limit cycle.

Ⓔ With respect to the previous case, equilibria e_1 and e_2 have somehow exchanged their role, both topologically (now e_1 lies at the left of e_2) and dynamically (e_1 is now a stable focus and no longer a saddle). An unstable limit cycle now bounds the basin of attraction around e_1 , in accordance to what pointed out in [18]. The time period of the cycle varies between zero and infinity. Actually, just after the Bogdanov–Takens bifurcation the limit cycle is tiny and has small period. When the subsequent homoclinic bifurcation is about to occur, the period of the limit cycle approaches infinity (see state portrait *F*).

Ⓕ Homoclinic bifurcation. The stable and unstable manifolds of the saddle e_2 join each other. The limit cycle has disappeared.

Ⓖ The size of the basin of attraction, which is again unbounded on one side, increases with δ. The portrait is not much different from the one depicted in Ⓒ.

6.3.1.4 Case UN_3

($\xi = 0.386, \mu_1 = 0.7, \mu_2 = 0.9, B_1{=}B_2 = -40, C_1 = 1.3, C_2 = 1.6, E_1 = 0, E_2 = 0.2$)
The only bifurcation taking place in this case is a heteroclinic one (Fig. 6.9).

Ⓐ Three equilibria can be found, one is a stable node, the others are saddles (as in case UN_1). The stable manifolds of the two saddles e_1 and e_3 bound the basin of attraction.

Ⓑ The basin of attraction is reduced and is now bounded by the stable manifolds of e_1 only.

6.3.2 *Oversteering Cases*

6.3.2.1 Case OV_0

($\xi = 0.386, \mu_1 = 0.9, \mu_2 = 0.7, B_1 = B_2 = -10, C_1 = C_2 = 1, E_1 = E_2 = 0$)

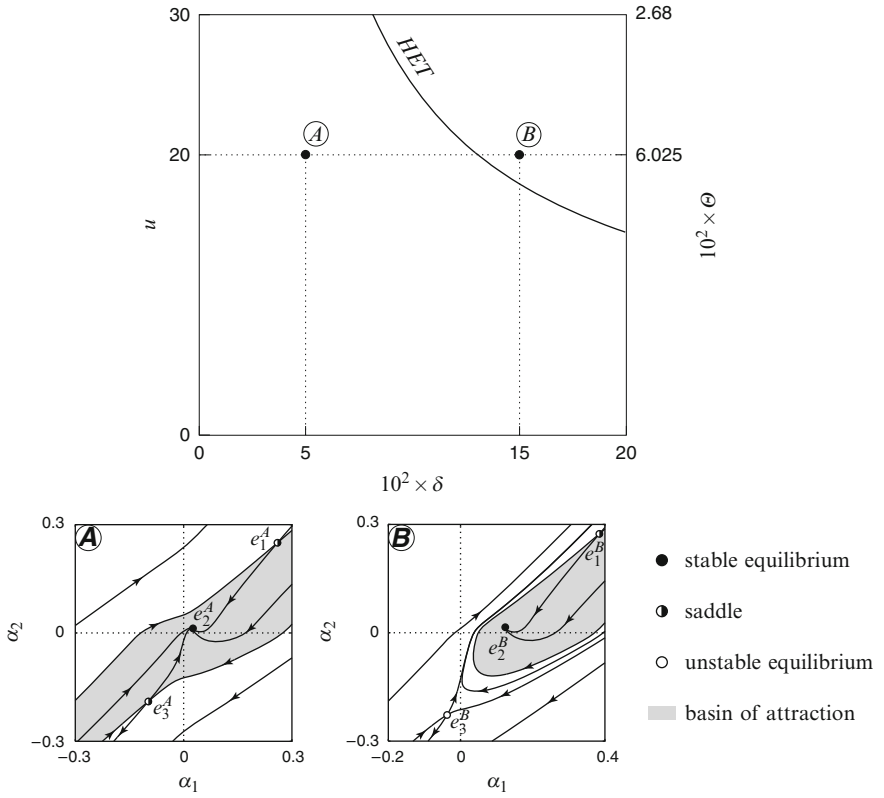

Fig. 6.9 Bifurcation diagram of the single-track model of Fig. 6.4 with respect to the inputparameters (δ, Θ)(case UN_3).$\xi = 0.386$. u is the forwardspeed, $\Theta = 24.1/u^2$. Thephase portraits Ⓐ and Ⓑ correspondto the (δ, Θ)values reported in the bifurcation diagram (*left panel*)

Three equilibria are possible for suitable (δ, Θ) values. Two of them are saddles (e_1 and e_3) and the other is a stable node (e_2) (Fig. 6.10).

Ⓐ, Ⓒ The boundary of the basin of attraction of the stable node e_2 is formed by the stable manifolds of the saddles e_1 and e_3. Case Ⓒ represents a less dangerous situation than case Ⓐ because the stable equilibrium ${e_2}^A$ is next to the border of the basin of attraction.

Ⓑ, Ⓓ The stable equilibrium e_2 and the saddle e_1 collide and disappear in a saddle-node bifurcation $SN_{1,2}$. No stable behaviour is possible for (δ, Θ) values above the bifurcation curve.

At zero steering angle δ, if Θ is decreased (i.e. the forward speed u is increased), a different bifurcation occurs between Ⓒ and Ⓓ. It is a subcritical pitchfork bifurcation, in which the two saddles e_1 and e_3 collide with the stable equilibrium

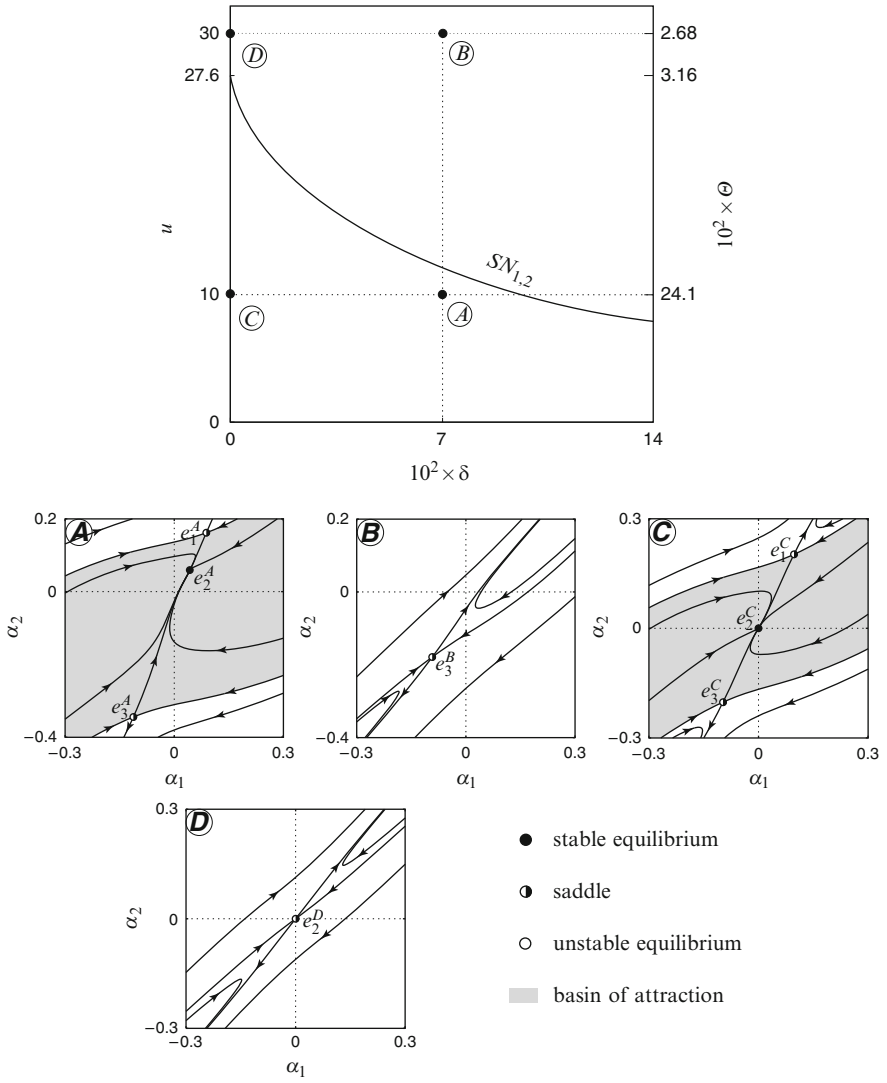

Fig. 6.10 Bifurcation diagram of the single-track model of Fig. 6.4 with respect to the input parameters (δ, Θ) (case OV_0). $\xi = 0.386$. u is the forward speed, $\Theta = 24.1/u^2$. The phase portraits Ⓐ to Ⓓ correspond to the (δ, Θ) values reported in the bifurcation diagram (*upper-left panel*)

e_2, disappear and leave e_2, unstable, as the only equilibrium. Such a bifurcation is well known by engineers. Actually it corresponds to the unstable straight running of oversteering vehicles [16].

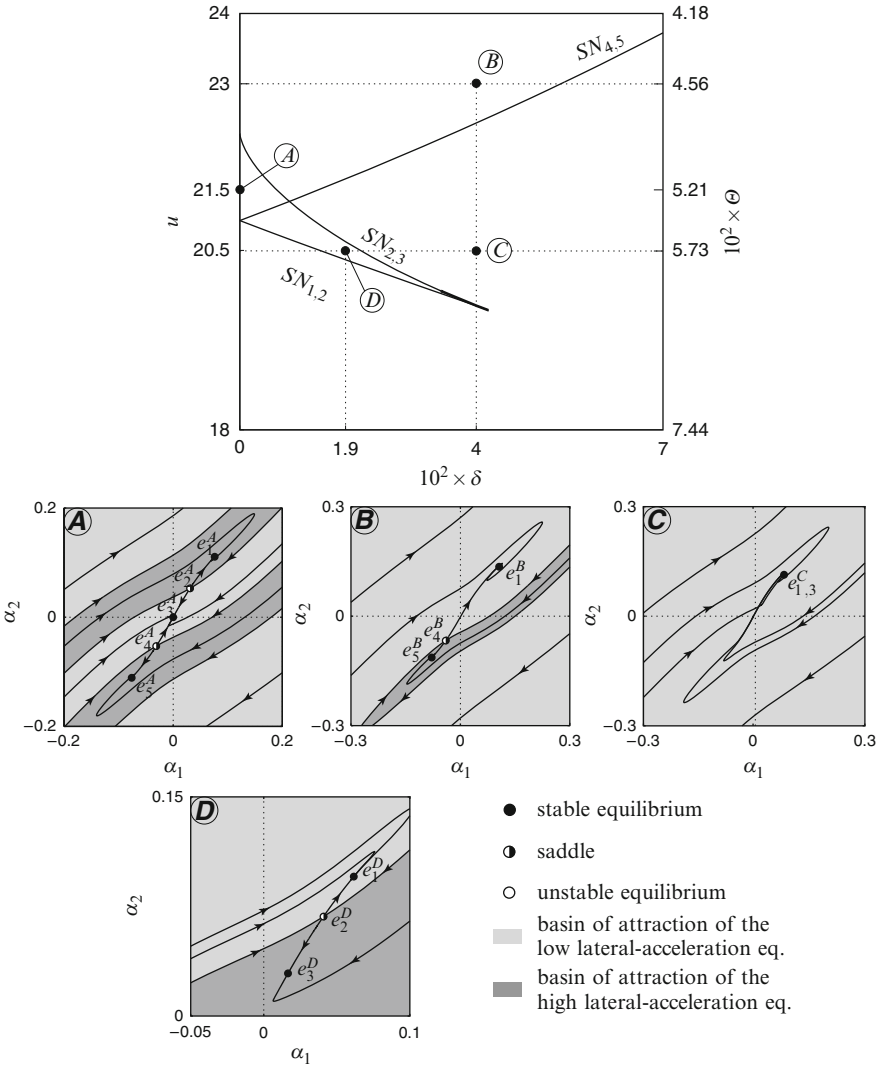

Fig. 6.11 Bifurcation diagram of the single-track model of Fig. 6.4 with respect to the input parameters (δ, Θ) (case OV_1). $\xi = 0.386$. u is the forward speed, $\Theta = 24.1/u^2$. The phase portraits Ⓐ to Ⓓ correspond to the (δ, Θ) values reported in the bifurcation diagram (*upper-left panel*)

6.3.2.2 Case OV_1

($\xi = 0.386, \mu_1 = 0.9, \mu_2 = 0.8, B_1 = -10, B_2 = -20, C_1 = C_2 = 1, E_1 = E_2 = 0$)

Three saddle-node bifurcations are found (Fig. 6.11). Multiple stable equilibria are present: this implies that, unlike the previous cases, it is possible to negotiate

different bends (up to three) at a given combination of speed (Θ) and steering angle δ. Stable equilibria at high lateral acceleration (i.e., at high lateral slips α_i (i = 1,2)) can exist. It may even happen that the vehicle is unstable for straight running, but is stable at high lateral acceleration. This situation is extremely interesting since is quite less intuitive. The strength of bifurcation analysis is again demonstrated in this example, that clarifies how clumsy could be performing driving tests without supporting them by a sound mathematical background.

Ⓐ Surprisingly, five equilibria have been found: two of them are saddles (e_2, e_4) and three (e_1, e_3, e_5) are stable ones. The three basins of attraction cover the entire phase space, which implies that no perturbation gives rise to unstable behavior. The stable manifolds of the saddles form the boundaries of the basin of attraction. Although the system never diverges to infinity, jumping from a stable equilibrium to another could actually cause problems to the driver in controlling the path of the vehicle. No analyzes of such a case can be found in the literature. The interpretation of the behavior of the car in this case is still a major research issue.

Ⓑ The saddle e_2 and the stable equilibrium e_3 collide and disappear at the saddle-node bifurcation $SN_{2,3}$. A catastrophic transition occurs, however trajectories do not diverge to infinity but remain bounded. They are captured by the basin of attraction of one of the other stable equilibria.

Ⓒ In this case a single stable equilibrium is present. Surprisingly, if the steering angle δ is sufficiently large, decreasing Θ (i.e., increasing the forward speed u) does not cause any catastrophe. Indeed, for $\delta \geq 4.3$ the equilibrium $e_{1,3}$ does not undergo any bifurcation (although its basin of attraction changes).

Ⓓ In this case two stable equilibria exist, both with positive coordinates (α_1, α_2). Surprisingly they correspond to bends negotiated in the same direction but with different radii. Such a situation can be observed in the actual tests, even if often it is taken by a measurement error rather than a matter of fact.

Let us consider the case in which, at $\Theta = 0.057$ $(u = u^D = u^C = 20.5)$, the steering angle is increased from $\delta = 0$ to $\delta = \delta^C = 4$ (see the upper-left panel of Fig. 6.11). At $\delta = 0$, e_3 is the only equilibrium and it is globally stable. Two more equilibria appear (e_1, e_2) if δ is increased. If the bifurcation curve $S_{1,2}$ is crossed, the driver does not feel anything because e_3 is unaffected by the bifurcation, however its global stability is lost. If δ is further increased, when the bifurcation curve $SN_{2,3}$ is crossed the equilibrium e_3 suddenly disappears, giving rise to a rapid transient with the car negotiating a tighter bend.

6.3.2.3 Case OV_2

$(\xi = 0.386, \mu_1 = \mu_2 = 0.9, B_1 = -14, B_2 = -10, C_1 = C_2 = 1.3, E_1 = -8, E_2 = -5)$

A stable limit cycle can be found for suitable (δ, Θ) combinations. The region in which such a limit cycle exists is bounded by a supercritical Hopf bifurcation curve (H) and a homoclinic bifurcation curve (HOM). Moreover, a saddle-node bifurcation is present (Fig. 6.12).

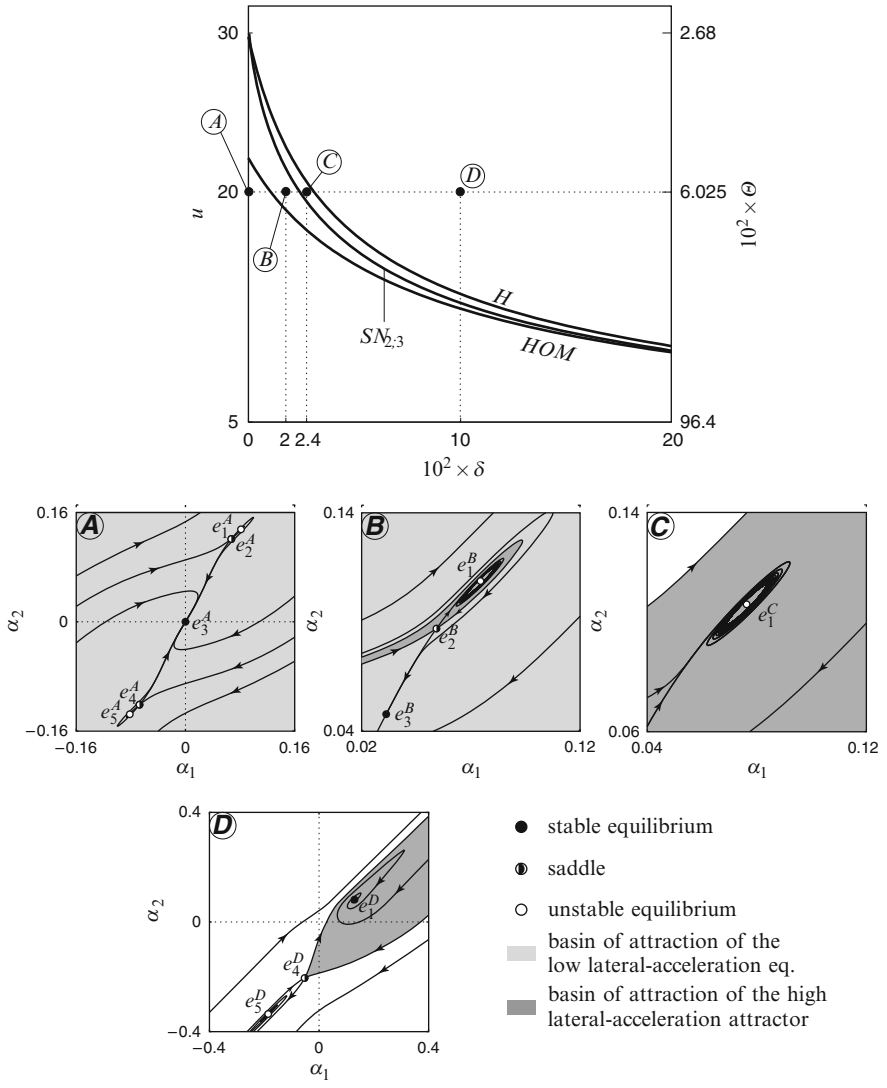

Fig. 6.12 Bifurcation diagram of the single-track model of Fig. 6.4 with respect to the input parameters (δ, Θ) (case OV_2). $\xi = 0.386$. u is the forward speed, $\Theta = 24.1/u^2$. The phase portraits Ⓐ to Ⓓ correspond to the (δ, u) values reported in the bifurcation diagram (*upper-left panel*)

Ⓐ Five equilibria are present: four of them are unstable (e_1 and e_5 are foci, e_2 and e_4 are saddles) and one is stable (e_3). Actually, the only stable equilibrium is globally stable, i.e., any perturbation is absorbed.

Ⓑ The homoclinic bifurcation gives birth to a stable limit cycle, whose basin of attraction is delimited by the two branches of the stable manifold of the saddle e_2.

Ⓒ At the $SN_{2,3}$ bifurcation, the stable equilibrium e_3 collides with the saddle e_2 and disappear. The only possible stable behaviour is now oscillatory, namely the

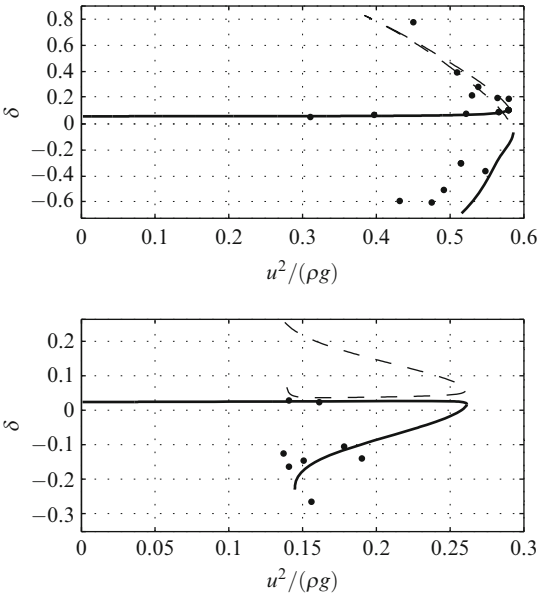

Fig. 6.13 Experimental observations of the steady-state steering angle δ as a function of the lateral acceleration $u^2/(\rho g)$. *Upper panel*: an SUV on a wet circular track ($\rho = 50$ m). *Lower panel*: a sports car on an icy circular track ($\rho = 100$ m). Multiple equilibria co-exist at large values of the lateral acceleration (adapted from [8])

stable limit cycle. The amplitude of the oscillation is considerable, being equal to $0.034\ rad = 2\ deg$. This behaviour has been occasionally reported by professional drivers as a very dangerous situation.

⓪ The stable limit cycle disappears by shrinking on the unstable focus e_1 at the supercritical Hopf bifurcation. Now e_1 becomes stable and its basin of attraction, which is the same of the former limit cycle, is bounded by the manifolds of the saddle e_4. Notice that in [16] some conditions for checking the local stability of an equilibrium are reported. In principle, using those conditions it is possible to reveal the Hopf bifurcation, but it is by no means possible to assess whether the limit cycle is stable.

6.3.3 Bifurcations and Multistability: Experimental Evidences

After the works of Mitschke [14] and Pacejka [16], it is generally accepted that the vehicle model presented in this chapter is not only qualitatively but also quantitatively correct, provided the steady-state lateral acceleration is low. This is the case, for example, at steady-state cornering on iced or snowy surface. As a matter of fact, in such conditions tyre nonlinearities influence the dynamic behavior very much, actually much more than suspension characteristics, which are not considered in the presented model.

An example of experiments reporting bifurcations and multistability in cornering vehicles can be found in [8], where the issue of multiple equilibria at limit lateral accelerations is addressed both theoretically and experimentally. Figure 6.13 shows the steering angle δ as a function of the steady-state lateral acceleration $u^2/(\rho g)$, for

two different vehicles: an SUV on a wet circular track ($\rho = 50$ m), and a sports car on an icy circular track ($\rho = 100$ m). Note that a complementary approach is adopted in our treatment, since in our bifurcation analysis we fix (δ, u) and study the (possibly multiple) steady-state values of ρ. But, apart from this, both panels clearly show that, at steady-state cornering, multiple equilibria (i.e., multiple δ values) are possible for a given velocity. This is fully consistent with the theoretical concepts exposed in this chapter, and allows one to claim that the tools of nonlinear dynamical systems analysis are fundamental for a fully understanding of the dynamic behavior of road vehicles running at limit conditions (see [7, 10, 19] for other contributions on this topic).

6.4 Conclusions

A general approach for describing the non linear dynamical behavior of a vehicle cornering on an even road surface has been presented. The bifurcation analysis of a two-degree-of-freedom model has been performed to provide an overview of the different behaviors of two-axle vehicles.

The equations of motion have been derived by taking into account all the important features of the vehicle. The non linear equations have been put in non dimensional form, allowing to reduce to a minimum the number of parameters needed to describe the vehicle behavior. Two parameters have been used to describe the running conditions (or driver inputs). They are the steering angle δ and a parameter Θ, which is related to the forward speed u and contains the wheelbase and the gravitational acceleration. Another non dimensional parameter, ξ, is related to the vehicle architecture and depends both on the position of the centre of mass and on the wheelbase. The non linear characteristics of the front and rear tyres are defined by a total of eight non dimensional parameters. Both the mass and the moment of inertia around a vertical axis passing by the centre of gravity do not appear in the non dimensional form of the equations of motion, and thus do not play any role in the non linear dynamic behavior of the car model adopted for this study. This conclusion seems quite original and important, because the model adopted in this chapter is considered reliable for a number of engineering applications concerning road vehicles (in particular automobiles).

Seven different combinations of front and rear tyre characteristics have been considered, corresponding to four understeering and three oversteering vehicles. For each case, a bifurcation analysis has been conducted by varying the two driver inputs, namely the front wheel steering angle δ and the parameter Θ related to the vehicle forward speed. The results are given in non dimensional form and thus they are able to represent a wide range of different vehicles. Precisely, they represent all of the vehicles having a peculiar architecture defined by $\xi = 0.386$. Such a value is typical to the vast majority of automobiles, at least at a given load configuration.

An impressive variety of different possible dynamic behaviors appears by changing the non linear tyre characteristics even slightly. If the vehicle is always

understeering, no bifurcations occur and unstable steady-state solutions never appear. This is a very good occurrence for active safety. Bifurcations occur if the forward speed is relatively high. Usually, by increasing the steady-state steering angle we obtain an increase of the Θ value at which a bifurcation occurs, to which a decrease in the speed corresponds. Heteroclinic, homoclinic, saddle-node, and Bogdanov–Takens bifurcations have been reported. An unstable limit cycle has been also found.

The oversteering bifurcation diagram are much more involved than the understeering ones. The same considerations introduced for the undesteering vehicle hold. Heteroclinic, homoclinic, saddle-node and Hopf bifurcations have been reported. Stable limit cycles have been also found. Extremely complex situations may occur, actually up to five different simultaneous equilibria have been found. Especially when multiple attractors are present, the vehicle can have different stable running conditions for the same driver inputs.

The above results hold under the hypothesis of small values of both the lateral slips and the steering angle. A further investigation could be performed in the future by removing such hypotheses and considering the full spin-out. No longitudinal forces at the tyres have been introduced and the body roll motion as well as the elastokinematics of the axles have been neglected. However, the different combinations of tyre characteristics taken into consideration provide—at least qualitatively—information on the effects of the neglected issues.

The bifurcation analysis illustrated in this chapter is very informative for chassis designers and control engineers. Any robust design of either the chassis or the stability controls should take into account the different cases that may arise referring to the dynamic behavior of the vehicle model presented in this chapter. The non dimensional form of the output of the bifurcation analysis provides a sound generality to the presented results.

6.5 Appendix: Degenerate Bogdanov–Takens Bifurcation

In this Appendix we study the normal form of the degenerate Bogdanov–Takens bifurcation found in Sect. 6.3.1.3. Notice that the unfolding of this bifurcation is not present in the literature, to the best of our knowledge, and it is only given as an exercise in the book by Carr [5]. Starting from there, we will develop the normal form of this bifurcation, to which system (6.5) can be reduced on the *BT* bifurcation curve through the standard technique of parameter-dependent normal form reduction [3]. From the analysis, we will understand why this doubly-degenerate bifurcation appears instead in our system as a codimension-one bifurcation.

Let us consider the second-order ODE system

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= p_1 x_1 + p_2 x_2 - {x_1}^2 - s x_1 x_2, \end{aligned} \tag{6.6}$$

where $s = \pm 1$, and let us study the system for $p = (p_1, p_2) \in \mathbb{R}^2$.

As a first step, we find that the system has always the two equilibria

$$e_1 = (0,0), \qquad e_2 = (p_1, 0).$$

The Jacobian of the system is

$$J(x,p) = \begin{pmatrix} 0 & 1 \\ -2x_1 - sx_2 + p_1 & -sx_1 + p_2 \end{pmatrix},$$

that, evaluated at the equilibria, takes values

$$J|_{e_1} = \begin{pmatrix} 0 & 1 \\ p_1 & p_2 \end{pmatrix}, \qquad J|_{e_2} = \begin{pmatrix} 0 & 1 \\ -p_1 & -sp_1 + p_2 \end{pmatrix}.$$

From the study of the linearized system, we can conclude that the origin e_1 is

- A stable node or focus, if $p_1 < 0$ and $p_2 < 0$,
- An unstable node or focus, if $p_1 < 0$ and $p_2 > 0$,
- A saddle, if $p_1 > 0$, $\forall p_2$.

Similarly, we can say for e_2 that it is

- A stable node or focus, if $p_1 > 0$ and $p_2 < sp_1$,
- An unstable node or focus, if $p_1 > 0$ and $p_2 > sp_1$,
- A saddle, if $p_1 < 0$, $\forall p_2$.

From this analysis we can derive the local bifurcations of the system, namely

TC: at $p_1 = 0$, e_1 and e_2 undergo a transcritical bifurcation,

H_{e_1}: at $p_2 = 0$and $p_1 < 0$, e_1 undergoes a Hopf bifurcation,

H_{e_2}: at $p_2 = sp_1$ and $p_1 > 0$, e_2 undergoes a Hopf bifurcation.

Let us now analyze in detail the Hopf bifurcations. At the Hopf bifurcation of equilibrium e_1, the Jacobian becomes

$$J|_{e_1} = \begin{pmatrix} 0 & 1 \\ p_1 & 0 \end{pmatrix},$$

so that, through the transformation

$$z = [z_1, z_2] = T[x_1, x_2], \qquad \text{with} \qquad T^{-1} = \begin{bmatrix} 0 & 1 \\ \sqrt{|p_1|} & 0 \end{bmatrix},$$

system (6.6) becomes

$$\dot{z}_1 = \sqrt{|p_1|}z_2 - sz_1z_2 - \frac{1}{\sqrt{|p_1|}}z_2^2,$$
$$\dot{z}_2 = \sqrt{|p_1|}z_1$$

The first Lyapunov coefficient [11] is therefore given by

$$l_1^{e_1} = \frac{1}{16\omega} f_{z_1,z_2}^{(1)} f_{z_2^2}^{(1)} = \frac{1}{16\sqrt{|p_1|}}(-s)\left(-\frac{1}{\sqrt{|p_1|}}\right) = \frac{s}{16|p_1|}.$$

The same computations can be done for e_2, and this lead us to the conclusion that both the Hopf bifurcations are subcritical in the case $s > 0$ and supercritical otherwise.

Let us analyze the case in which the Hopf bifurcations are subcritical ($s > 0$), as in the case described in Sect. 6.3.1.3. Trough normal form analysis, we know that two homoclinic bifurcation curves lie in the parameter half-plane in which the saddle has positive saddle quantity (e_1 if $p_1 < 0$, e_2 if $p_1 > 0$), since the limit cycle is unstable. As for the standard Bogdanov–Takens case (e.g., [11], p. 395), in order to prove it we have to reduce our system to a perturbation of a Hamiltonian system. First, we consider the homoclinic bifurcation of the non-trivial equilibrium e_2. Let us perform a singular rescaling to make the distance between the equilibria equal to 1, independently of the parameters ("blowing-up"), and let us introduce a new time variable τ. More in detail:

$$z_1 = \zeta_1 p_1, \qquad z_2 = -\zeta_2 p_1^{3/2}, \qquad t = -\tau\sqrt{p_1}.$$

With these substitutions, system (6.6) becomes

$$\begin{aligned} \dot{\zeta}_1 &= \zeta_2, \\ \dot{\zeta}_2 &= \zeta_1(\zeta_1 - 1) - (\gamma_1\zeta_2 + \gamma_2\zeta_1\zeta_2), \end{aligned} \tag{6.7}$$

where the dots now means derivative with respect to τ, and we have introduced the two new parameters

$$\gamma_1 = \frac{p_2}{\sqrt{p_1}}, \qquad \gamma_2 = -\sqrt{p_1}. \tag{6.8}$$

Near the codimension-two point, system (6.7) is a perturbation of the Hamiltonian system

$$\dot{\zeta}_1 = \zeta_2,$$
$$\dot{\zeta}_2 = \zeta_1(\zeta_1 - 1),$$

with Hamilton function

$$H(\zeta_1, \zeta_2) = \frac{\zeta_1^2}{2} + \frac{\zeta_2^2}{2} - \frac{\zeta_1^3}{3}.$$

The homoclinic horbit passes trough the saddle, and so does the level curve of the Hamilton function with value $H(1, 0) = 1/6$. Without going into all the details (we refer the interested reader to [11]), we can define a test-function (*split function*) such that the homoclinic bifurcation exists when $\|\gamma\| > 0$ (i.e., in the non-hamiltonian case). By means of this split function, it is possible to prove the uniqueness and the hyperbolicity of the bifurcating limit-cycle (this implies that no other bifurcation exists). Moreover, the asymptotic form of the homoclinic curve HOM_{e_2} is

$$\mathrm{HOM}_{e_2} = \left\{ (\gamma_1, \gamma_2) : \gamma_1 = -\frac{1}{7}\gamma_2 + o(|\gamma_2|), \gamma_2 \geq 0 \right\}.$$

Using (6.8), we can obtain the asymptotic form of the homoclinic bifurcation curve in the original parameter space, that is

$$\mathrm{HOM}_{e_2} = \left\{ (p_1, p_2) : p_2 = \frac{1}{7}p_1, p_1 \leq 0 \right\}.$$

In order to find the asymptotic expansion of the other homoclinic orbit, we first translate our system in order to move the equilibrium e_2 (non-saddle) to the origin of the state space, by the translation

$$\eta_1 = z_1 - p_1, \qquad \eta_2 = z_2.$$

Then, through the following change of coordinates and time rescaling

$$\eta_1 = -\zeta_1 p_1, \qquad \eta_2 = -\zeta_2 p_1^{3/2}, \qquad t = \tau\sqrt{p_1},$$

and the parameter transformation

$$\gamma_1 = \frac{p_1 - p_2}{\sqrt{p_1}}, \qquad \gamma_2 = -\sqrt{p_1}, \tag{6.9}$$

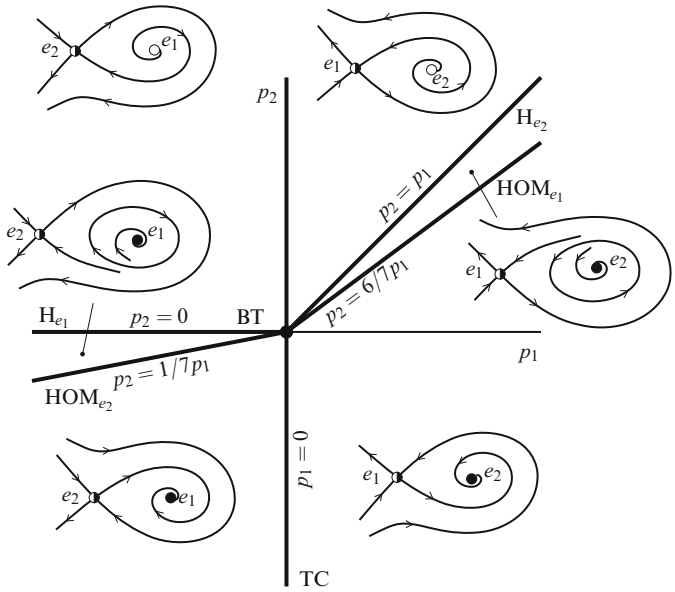

Fig. 6.14 Bifurcation diagram of the normal form (6.6) in the case $s = 1$

we reduce again our system to (6.7). At this point, by repeating the above analysis, we obtain the asymptotic formula of the second homoclinic curve which, in the original parameters, reads

$$\mathrm{HOM}_{e_1} = \left\{ (p_1, p_2) : p_2 = \frac{6}{7} p_1, p_1 \geq 0 \right\}.$$

We are now in the condition to sketch the bifurcation diagram in the neighborhood of the codimension-two point, as reported in Fig. 6.14. The bifurcation found in Sect. 6.3.1.3 corresponds to passing from the region between the curves(HOM_{e_2}, TC) to the region between the curves(H_{e_2},HOM_{e_1}). Thanks to the fact that all the bifurcation curves are asymptotically linear, with different slopes, the above described passage is generically possible, provided the original parameters are a suitable combination of (p_1, p_2).

References

1. Andrzejewski, R., Awrejcewicz, J.: Nonlinear Dynamics of a Wheeled Vehicle. Springer (2005)
2. Arctic Falls Proving Grounds: http://www.arcticfalls.se/. Accessed 30 May 2012
3. Beyn, W.J., Champneys, A.R., Doedel, E.J., Govaerts, W., Kuznetsov, Yu.A., Sandstede, B.: Numerical continuation, and computation of normal forms. In: B. Fiedler (ed.) Handbook of

Dynamical Systems, vol. 2, 3rd edn., pp. 149–219. American Mathematical Society, Burlington, MA (2002)
4. Bruni, S., Mastinu, G. (eds.): Proceedings of the 19th IAVSD Symposium: Dynamics of vehicles on roads and tracks. Milan, Italy (2005)
5. Carr, J.: Application of Centre Manifold Theory. Springer (1981)
6. Catino, B., Santini, S., Bernardo, M.D.: Mcs adaptive control of vehicle dynamics: an application of bifurcation techniques to control system design. In: Proceedings of the 42nd IEEE Conference on Decision and Control, pp. 2252–2257. Maui, Hawaii, USA (2003)
7. Edelmann, J., Ploechl, M.: Handling characteristics and stability of the steady-state powerslide motion of an automobile. Regular & Chaotic Dynamics **14**(6), 682–692 (2009)
8. Edelmann, J., Ploechl, M., Pfeffer, P.: Analysis of steady-state vehicle handling and driver behaviour at extreme driving conditions. In: Proceedings of the 22th IAVSD Symposium: Dynamics of vehicles on roads and tracks. Manchester, UK (2011)
9. Hedrick, K. (ed.): Proceedings of the 20th IAVSD Symposium: Dynamics of Vehicles on Roads and Tracks. Berkeley, CA, USA (2007)
10. Hindiyeh, R.Y., Gerdes, J.C.: Equilibrium analysis of drifting vehicles for control design. In: Proceedings of the Asme Dynamic Systems and Control Conference 2009, Hollywood, CA, 2009, pp. 181–188 (2009)
11. Kuznetsov, Yu.A.: Elements of Applied Bifurcation Theory, 3rd edn. Springer-Verlag, Berlin (2004)
12. Liaw, D.C., Chung, W.C.: A feedback linearization design for the control of vehicle's lateral dynamics. Nonlinear Dynamics **52**, 313–329 (2008)
13. Mastinu, G., Cesarini, R.: Stability and drivability of vehicles in the horizontal plane. ATA **44**, 430–442 (1991). In Italian
14. Mitschke, M., Wallentowitz, H.: Dynamik der Kraftfahrzeuge, 4rd edn. Springer, Berlin (2004). In German
15. Orosz, G., Krauskopf, B., Wilson, R.: Bifurcations and multiple traffic jams in a vehicle-following model with reaction-time delay. Physica D **211**, 277–293 (2005)
16. Pacejka, H.: Tire and Vehicle Dynamics, 4rd edn. Elsevier (2006)
17. Palkovics, L., Venhovens, P.: Investigation on stability and possible chaotic motions in the controlled wheel suspension system. Vehicle System Dynamics **21**, 269–296 (1992)
18. Shen, S., Wang, J., Shi, P., Premier, G.: Nonlinear dynamics and stability of vehicle plane motions. Vehicle System Dynamics **45**, 15–35 (2007)
19. Voser, C., Hindiyeh, R.Y., Gerdes, J.C.: Analysis and control of high sideslip manoeuvres. Vehicle System Dynamics **48**(1), 317–336 (2010)
20. Zhu, Q., Ishitobi, M.: Chaos and bifurcations in a non linear vehicle model. Journal of Sound and Vibration **275**, 1136–1146 (2004)

Chapter 7
Detecting and Exploiting Chaotic Transport in Mechanical Systems

Shane D. Ross and Phanindra Tallapragada

7.1 Introduction

Several geometric and probabilistic methods for studying chaotic phase space transport have been developed and fruitfully applied to diverse areas from orbital mechanics, chemistry, biomechanics to fluid mechanics and beyond [1–31]. There is much interest in understanding how chaos arises in dynamical systems, how to detect it, how to exploit it when it is present, and how to design for (or against) it [32]. Increasingly, systems of interest are determined not by analytically defined model systems, but by data from experiments or large-scale simulations from many areas of the physical sciences, including atomic physics [4], geophysical fluid dynamics [33–37], musculoskeletal biomechanics [38–45], and space mission design [1, 46–51]. We will discuss those features of chaotic phase space transport in finite-time, non-autonomous systems which seem to be robust, considering invariant manifolds and invariant manifold-like objects, and their connection with concepts such as symbolic dynamics [1], braids [52–55], chaos, almost-invariant sets [18, 56], and coherent sets [22].

These methods fall into two main categories, the geometric and the probabilistic. Under the umbrella of geometric methods are the techniques of invariant manifolds (of fixed points or larger invariant sets) [1–4, 57], lobe dynamics [18, 37, 58, 59], finite-time Lyapunov exponents (FTLE) and Lagrangian coherent structures (LCS) [35, 60–65]. The method of FTLE-LCS has proven to be particularly useful in studying transport in time-dependent systems and has found a variety of

S.D. Ross (✉)
Engineering Science and Mechanics, Virginia Tech, Blacksburg, VA, USA
e-mail: sdross@vt.edu

P. Tallapragada
Mechanical Engineering and Engineering Science, University of North Carolina at Charlotte, NC, USA
e-mail: phanindra@uncc.edu

S. Banerjee et al. (eds.), *Applications of Chaos and Nonlinear Dynamics in Science and Engineering - Vol. 2*, Understanding Complex Systems,
DOI 10.1007/978-3-642-29329-0_7,

applications [25–30]. The probabilistic approach studies the transport of densities and measures and the so-called almost invariant sets (AISs) and coherent sets. These methods too have been successfully applied in the study of various geophysical flow problems [22] and mixing in microchannels [31].

The development of the above two methods to non-autonomous settings has occurred almost simultaneously in the last decade. The geometric methods, particularly FTLE-LCS, study stretching and contraction around reference trajectories and identify transport barriers. The FTLE-LCS method is therefore a local method; it provides local transport information from which one attempts to infer the global transport properties of the system. The probabilistic method, on the other hand, ignores the local transport structures, but using the transfer operator divides the phase space into maximally invariant sets.

One engineering application we consider is the determination of optimal routes for weakly propelled immersed in complex environmental fluid flows, by developing control schemes which explicitly incorporate the geometry of the dynamical structures which organize the flow. We consider the methods we describe as a stepping stone to a formal linking between the theory of chaotic transport and optimal control.

7.2 Computing Chaotic Transport

The theory of transport in phase space is a unified mathematical description of dynamical processes that can be applied to a wide range of physical phenomena across many scales. A geometric picture describing transport is emerging for time-chaotic (aperiodic) flow fields [1–3, 18, 28, 41, 47–50, 66–75]. The key is to partition the phase space into subsets and then to identify how phase points travel within and between the subsets [18]. In time-periodic systems, the stable and unstable manifolds of normally hyperbolic invariant manifolds (NHIMs) [1, 2, 76]—e.g., fixed points, periodic orbits, or other bound sets or orbits—provide the co-dimension one separatrices.

7.2.1 Transport Between Regions in Phase Space

To make the description of phase space transport more precise, consider the following. Suppose we have a volume- and orientation-preserving map $f : M \to M$ (e.g., a time-T (stroboscopic) map ϕ_t^{t+T} or a Poincaré map resulting from a continuous time dynamical system) on some compact set $M \subset \mathbb{R}^n$ with volume-measure μ and we seek a suitable (i.e. depending on the application in mind) *partition* of M into compact *regions of interest* R_i, $i = 1, \ldots, N_R$, as shown schematically in Fig. 7.1, such that

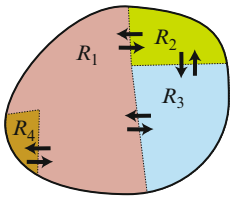

Fig. 7.1 The manifold M is partitioned into the regions R_i, $i = 1, \ldots, N_R$. If points are distributed uniformly over M at $t = 0$, we want to compute the movement of points between these regions for all times $t > 0$

$$M = \bigcup_{i=1}^{N_R} R_i \quad \text{and} \quad \mu(R_i \bigcap R_j) = 0 \text{ for } i \neq j. \tag{7.1}$$

Furthermore, we are interested in the following questions concerning the transport between the regions R_i (see [77]): In order to keep track of the initial condition of a point as it moves throughout the regions, we say that initially (i.e., at $t = 0$) region R_i is uniformly covered with species S_i. Thus, the species type of a point indicates the region in which it was located initially. Then we can generally state the transport problem as follows.

Describe the distribution of species S_i, $i = 1,\ldots,N_\mathrm{R}$, *throughout the regions* R_j, $j = 1,\ldots,N_\mathrm{R}$, *for any time* $t = n > 0$.

The quantity we want to compute is $T_{i,j}(n) \equiv$ *the total amount of species* S_i *contained in region* R_j *immediately after the n-th iterate*.

The *flux* $\alpha_{i,j}(n)$ of species S_i into region R_j on the n-th iterate is the change in the amount of species S_i in R_j on iteration n; namely, $\alpha_{i,j}(n) = T_{i,j}(n) - T_{i,j}(n-1)$. Since f is assumed volume-preserving, the flux is equal to the amount of species S_i entering region R_j at iteration n minus the amount of species S_i leaving R_j at iteration n.

Our goal is to determine $T_{i,j}(n)$, $i, j = 1, \ldots, N_R$ for all n. Note, that $T_{i,i}(0) = \mu(R_i)$, and $T_{i,j}(0) = 0$ for $i \neq j$. In the following we briefly describe the theoretical background behind the two computational approaches to the transport problem.

7.2.2 Lobe Dynamics

Lobe dynamics theory has been developed for the case $n = 2$ [58] and states that the two-dimensional phase space M of the map f can be divided as outlined above (see (7.1)). We define a region to be a connected subset of M with boundaries consisting of parts of the boundary of M (which may be at infinity) and/or segments of stable and unstable invariant manifolds of saddle-type hyperbolic fixed points, p_i, $i = 1, \ldots, N$, as shown schematically in Fig. 7.2a. When the manifolds $W^u(p_i)$ and $W^s(p_i)$ are followed out on a global scale, they often intersect in primary

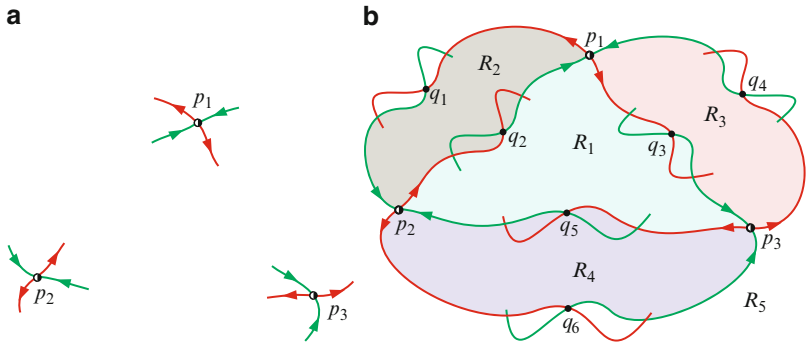

Fig. 7.2 (**a**) Pieces of the local unstable and stable manifolds, $W^u(p_i)$ (*red*) and $W^s(p_i)$ (*green*) of saddle fixed points $\{p_i\}$. (**b**) When the manifolds $W^u(p_i)$ and $W^s(p_i)$ are followed out on a global scale, they often intersect in primary intersection points $\{q_i\}$. These intersections allow one to define boundaries betweens regions $\{R_i\}$

intersection points $\{q_i\}$. These intersections allow one to define boundaries betweens regions $\{R_i\}$, as illustrated in Fig. 7.2b. Moreover, the transport between regions of phase space can be completely described by the dynamical evolution of small subregions of phase space, "lobes" enclosed by segments of the stable and unstable manifolds as defined below.

7.2.2.1 Boundaries, Regions, Pips, Lobes, and Turnstiles Defined

In order to define a *boundary* between regions, we must first define a *primary intersection point*, or *pip*. Consider just two saddle points, p_i and p_j. Suppose $W^u(p_i)$ and $W^s(p_j)$ intersect transversally at least once, i.e., $W^u(p_i) \bigcap W^s(p_j) \neq \emptyset$. Then by the definition of stable and unstable manifolds, they must intersect an infinite number of times. Consider just two of these intersection points, q and $\bar{q}$, as in Fig. 7.3. An intersection point r is a pip if $U[p_i, r]$ intersects $S[r, p_j]$ only at the point r, where $U[p_i, r]$ and $S[r, p_j]$ are segments of the unstable and stable manifolds, $W^u(p_i)$ and $W^s(p_j)$, respectively. In Fig. 7.3, q is a pip, but $\bar{q}$ is not a pip. The union of segments of the unstable and stable manifolds, which terminate in a pip, naturally form *boundaries* between *regions*. Considering Fig. 7.3b, define $B \equiv U[p_i, q] \bigcup S[p_j, q]$ as the boundary between "two sides," the regions R_1 and R_2. Note that we could have $p_i = p_j$.

Let $q_0, q_1 \in W^u(p_i) \bigcap W^s(p_j)$ be two adjacent pips, as in Fig. 7.4a, i.e., there are no other pips on $U[q_0, q_1]$ and $S[q_0, q_1]$, the segments of $W^u(p_i)$ and $W^s(p_j)$ connecting q_0 and q_1. We refer to the region interior (int) to $U[q_0, q_1] \bigcup S[q_0, q_1]$ as a *lobe*. Now consider again a pip q, as in Fig. 7.4b. Since q is a pip, $f^{-1}(q)$ is also a pip; both are part of the heteroclinic orbit passing through q, $h_q = \{\ldots, f^{-2}(q), f^{-1}(q), q, f^1(q), f^2(q), \ldots\}$ asymptotically going from p_i to p_j, i.e., $f^{-n}(q) \to p_i$ and $f^n(q) \to p_j$ as $n \to \infty$. As f is orientation-preserving, there is at least one pip on

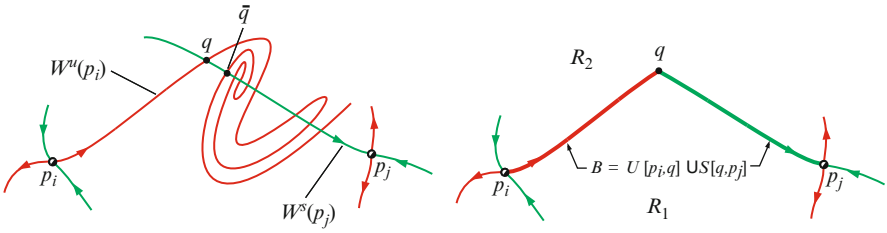

Fig. 7.3 (**a**) q is a primary intersection point (pip), while $\bar{q}$ is not. (**b**) Suppose $q \in W^u(p_i) \bigcap W^s(p_j)$ is a pip. We define $B \equiv U[p_i, q] \bigcup S[q, p_j]$ as a boundary between "two sides," R_1 and R_2

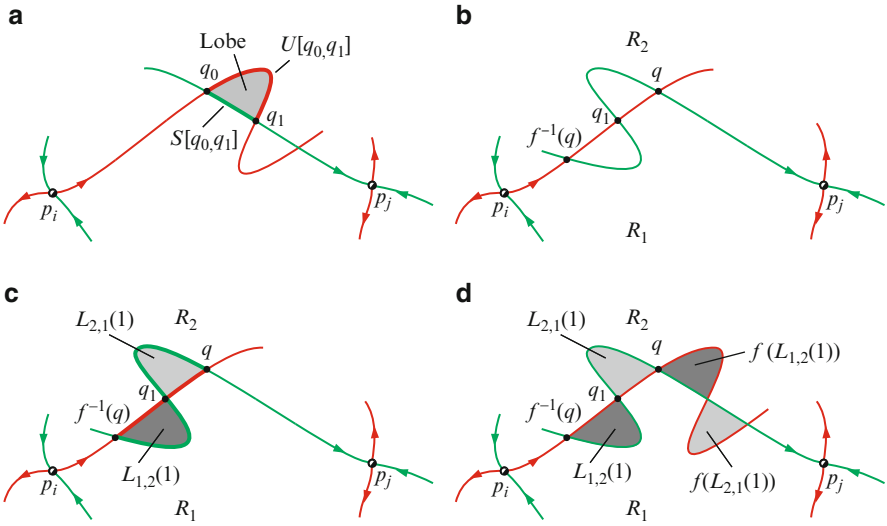

Fig. 7.4 (**a**) The region interior to $U[q_0, q_1] \bigcup S[q_0, q_1]$ is a lobe. (**b**) If q is a pip, so is $f^{-1}(q)$. Also, since f is orientation-preserving there is at least one pip, e.g., q_1, in $U[f^{-1}(q), q]$ where the $W^u(p_i)$, $W^s(p_j)$ intersection is topologically transverse. (**c**) $U[f^{-1}(q), q] \bigcup S[f^{-1}(q), q]$ forms the boundary of precisely two lobes; one in R_1, labeled $L_{1,2}(1)$, and the other in R_2, labeled, $L_{2,1}(1)$. (**d**) Under one iteration of f, only points in $L_{1,2}(1)$ can move from R_1 into R_2 by crossing B; and similarly for points in $L_{2,1}(1)$. The two lobes $L_{1,2}(1)$ and $L_{2,1}(1)$ are called a turnstile

$U[f^{-1}(q), q]$ where the $W^u(p_i)$, $W^s(p_j)$ intersection is topologically transverse. Let us assume there is only one such pip, called q_1, as illustrated in Fig. 7.4b; it belongs to a distinct heteroclinic orbit $h_{q_1} \neq h_q$ connecting p_i and p_j. Then $S[f^{-1}(q), q] \bigcup U[f^{-1}(q), q]$, highlighted in Fig. 7.4c, forms the boundary of precisely two lobes; one in R_1,

$$L_{1,2}(1) \equiv \text{int}(\text{U}[\text{f}^{-1}(\text{q}), \text{q}_1] \bigcup \text{S}[\text{f}^{-1}(\text{q}), \text{q}_1]) \tag{7.2}$$

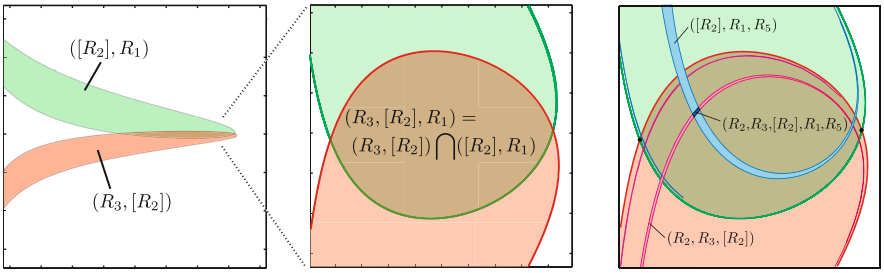

Fig. 7.5 Consider an example with several regions $\{R_1, R_2, R_3, R_4, R_5\}$, where we seek to label lobe intersections accordingly. (**a**) Consider two intersecting lobes $([R_2], R_1)$ and $(R_3, [R_2], R_1)$. (**b**) Zooming-in, denote the intersection $(R_3, [R_2]) \bigcap ([R_2], R_1)$ by $(R_3, [R_2], R_1)$. (**c**) Longer itineraries correspond to smaller pieces of phase space

and the other in R_2,

$$L_{2,1}(1) \equiv \text{int}(\text{U}[\text{q}_1, \text{q}] \bigcup \text{S}[\text{q}_1, \text{q}]). \tag{7.3}$$

Under one iteration of f, the only points that can move from R_1 into R_2 by crossing B are those in $L_{1,2}(1)$. The amount of phase space transported is $\mu(L_{1,2}(1)$. Similarly, under one iteration of f the only points that can move from R_2 into R_1 by crossing B are those in $L_{2,1}(1)$, where $\mu(L_{2,1}(1) = \mu(L_{1,2}(1)$; see Fig. 7.4d. In some cases, $\mu(L_{1,2}(1)$ can be computed from sums of action differences between the heteroclinic orbits h_q and h_{q1} [78].

The two lobes $L_{1,2}(1)$ and $L_{2,1}(1)$ are called a *turnstile* [79] and individually they are turnstile lobes. The essence of lobe dynamics can be stated as follows: the dynamics associated with crossing B is reduced to the dynamics of the turnstile lobes associated with B.

These results generalize to the boundary between any two sets R_i and R_j. We have chosen our notation such that $L_{i,j}(m)$, $m \geq 2$ denotes the lobe that leaves R_i and enters R_j on the m-th iterate, so that $L_{i,j}(m) = f^{1-m}(L_{i,j}(1))$. It is important to note that $L_{i,j}(m)$, $m \geq 2$, need not be contained entirely in R_i, and $f^n(L_{i,j}(1))$, $n \geq 2$, need not be contained entirely in R_j, i.e., the lobes can leave and re-enter regions with strong implications for the dynamics.

We can consider the lobes as prototypes of "mobile" subregions, "atoms" of phase space transport, whose images and pre-images under f will visit several regions. As an alternative notation, we can label these subregions based on their itineraries—that is, their past, present, and future whereabouts with respect to the regions $\{R_i\}$. For example, we can label $L_{1,2}(1)$ as $([R_1], R_2)$, where the bracketed region is the current region, and under one iterations of f, the pre-image will be entirely contained in R_2; thus, $f([R_1], R_2) = (R_1, [R_2])$. As in Fig. 7.5, itinerary labels of subregions can be naturally concatenated according to the labels of their corresponding intersected subregions. With this notation, we can label subregions with complicated itineraries, e.g., $(R_2, R_3, [R_2], R_1, R_5)$, where longer itineraries

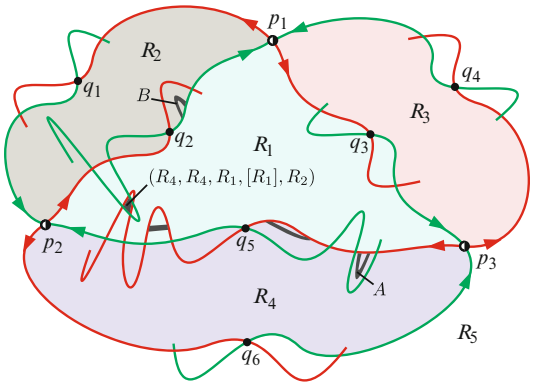

Fig. 7.6 The same regions as given in Fig. 7.2b are redrawn, this time extending some of the manifolds further to highlight the intersection of manifolds which bounds the subregion with itinerary $(R_4, R_4, R_1, [R_1], R_2)$. The image of this subregion under the map is B, i.e., $B = f(R_4, R_4, R_1, [R_1], R_2) = (R_4, R_4, R_1, R_1, [R_2])$, while $A = f^{-3}(R_4, R_4, R_1, [R_1], R_2) = ([R_4], R_4, R_1, R_1, R_2)$

will belong to smaller regions of phase space. In a later section, we will use these subregion itinerary labels to go between distant regions of the phase space using small controls, e.g., going from region R_4 to region R_1 in the schematic illustration of Fig. 7.6.

7.2.2.2 Multilobe, Self-Intersecting Turnstiles

Before going further, some comments regarding technical points are in order [58]. Above we assumed that there was only one pip between q and $f^{-1}(q)$, but this is not the case in all applications. Suppose that there are k pips, $k \geq 1$, along $U[f^{-1}(q), q]$ besides q and $f^{-1}(q)$. This gives rise to $k + 1$ lobes; m in R_2 and $(k + 1) - m$ in R_1. Suppose

$$L_0, L_1, \cdots, L_{k-m} \subset R_1,$$
$$L_{k-m+1}, L_{k-m+2}, \cdots, L_k \subset R_2.$$

Then we define

$$L_{1,2}(1) \equiv L_0 \bigcup L_1 \bigcup \cdots \bigcup L_{k-m},$$
$$L_{2,1}(1) \equiv L_{k-m+1} \bigcup L_{k-m+2} \bigcup \cdots \bigcup L_k,$$

and all the previous results hold.

An example of a multilobe turnstile comes from the study of orbits around rotating distended asteroids. Using the planar restricted full two-body problem as

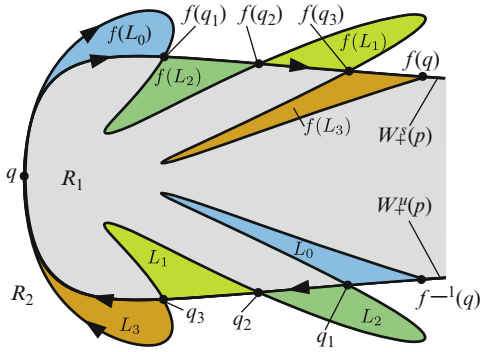

Fig. 7.7 An example of a multilobe turnstile with $k = 3$, $m = 2$. The system is the motion of a particle (e.g., ejecta) in the field of a rotating asteroid and is a Hamiltonian system. The coordinates shown here are (r, p_r), where r is the distance from the center of the asteroid and p_r is its conjugate momentum

a model (details provided in [71]), we can consider the trajectories of particles at a constant Hamiltonian energy, viewed on a 2D Poincaré surface-of-section. Particles can be ejected from the system (out to infinity) or captured into the system (from infinity). The phase space structure governing these phenomena is a homoclinic tangle associated with stable and unstable manifolds of a fixed point at infinity. Using (r, p_r) as coordinates along the surface-of-section, where r is the distance from the center of the asteroid and p_r is its conjugate momentum, this point is at $p = (\infty, 0)$. As shown in Fig. 7.7, the resulting homoclinic tangle has is a multilobe turnstile with $k = 3$ and $m = 2$.

Furthermore, we previously assumed that $L_{1,2}(1)$ and $L_{2,1}(1)$ lie entirely in R_1 and R_2, respectively. But $L_{1,2}(1)$ may intersect $L_{2,1}(1)$, as shown schematically in Fig. 7.8a for the case of a homoclinic tangle. We want $U[f^{-1}(q), q]$ and $S[f^{-1}(q), q]$ to intersect only in pips, so we must redefine our lobes, as shown in Fig. 7.8b. Let

$$I = \mathrm{int}\left(\mathrm{L}_{1,2}(1) \bigcap \mathrm{L}_{2,1}(1)\right).$$

The lobes defining the turnstile are redefined as

$$\begin{aligned} \tilde{L}_{1,2}(1) &\equiv L_{1,2}(1) - I, \\ \tilde{L}_{2,1}(1) &\equiv L_{2,1}(1) - I, \end{aligned} \tag{7.4}$$

and all our previous results hold. We note that multilobe, self-intersecting turnstiles [18] do indeed appear in physical systems, and may be a geometric phase space "constraint" which has a significant impact on the dynamics.

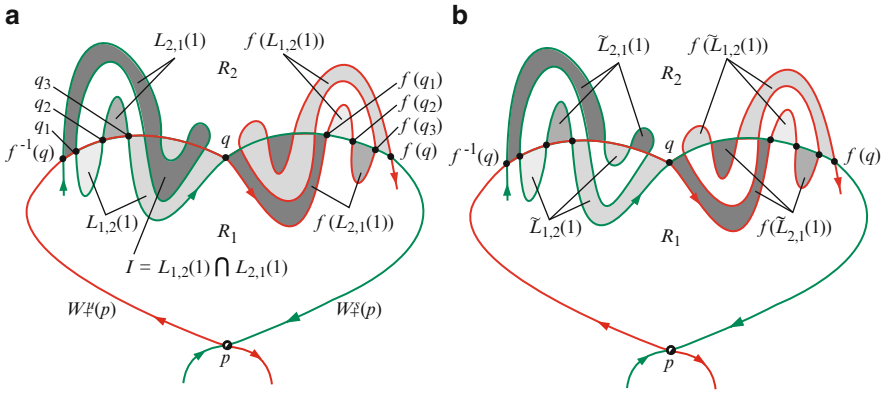

Fig. 7.8 *Comparison of FTLE-LCS and invariant manifolds in oscillating vortex pair flow* (**a**) Unstable (*red*) and stable (*black*) manifolds of the two fixed points near (± 1. 5, 0) for the system parameters (γ, ε) = (0. 5, 0. 1). See [83] for details. (**b**) FTLE-based LCS for the same system parameters. Notice that the ridge of high forward ($T > 0$) FTLE (*red*), the repelling LCS, corresponds with the stable manifold of the left fixed point of (**a**), and similarly for the high backward ($T < 0$) FTLE (*gray*, the attracting LCS, which corresponds with the unstable manifold of the right fixed point of (**a**). Computations by Shibabrat Naik

7.2.2.3 Expressions for the Transport of Species

Complicated transport properties can result from lobe dynamics (cf. [1, 58, 59, 78, 80]). Using the lobe dynamics framework, the transport of species between the regions—$T_{i,j}(n)$—can be computed via the following formulas (Fig. 7.9).

Let $L^k_{i,j}(m) \equiv L_{i,j}(m) \bigcap R_k$ denote the portion of lobe $L_{i,j}(m)$ that is in the region R_k. Then

$$T_{i,j}(n) - T_{i,j}(n-1) = \sum_{k=1}^{2} \left[\mu(L^i_{k,j}(n)) - \mu(L^i_{j,k}(n))\right] \tag{7.5}$$

where

$$\begin{aligned}\mu(L^i_{k,j}(n)) = &\sum_{s=1}^{2}\sum_{m=0}^{n-1} \mu\Big(L_{k,j}(1) \bigcap f^m(L_{i,s}(1))\Big) \\ &- \sum_{s=1}^{2}\sum_{m=1}^{n-1} \mu\Big(L_{k,j}(1) \bigcap f^m(L_{s,i}(1))\Big).\end{aligned} \tag{7.6}$$

Thus, the dynamics associated with crossing a boundary B is reduced completely to a study of the dynamics of the turnstile lobes associated with B. The amount of computation necessary to obtain all the $T_{i,j}(n)$ can be reduced due to conservation of area and species, as well as symmetries of the map f. In some cases, one can obtain A typical species transport calculation result is given in Fig. 7.10 where only two

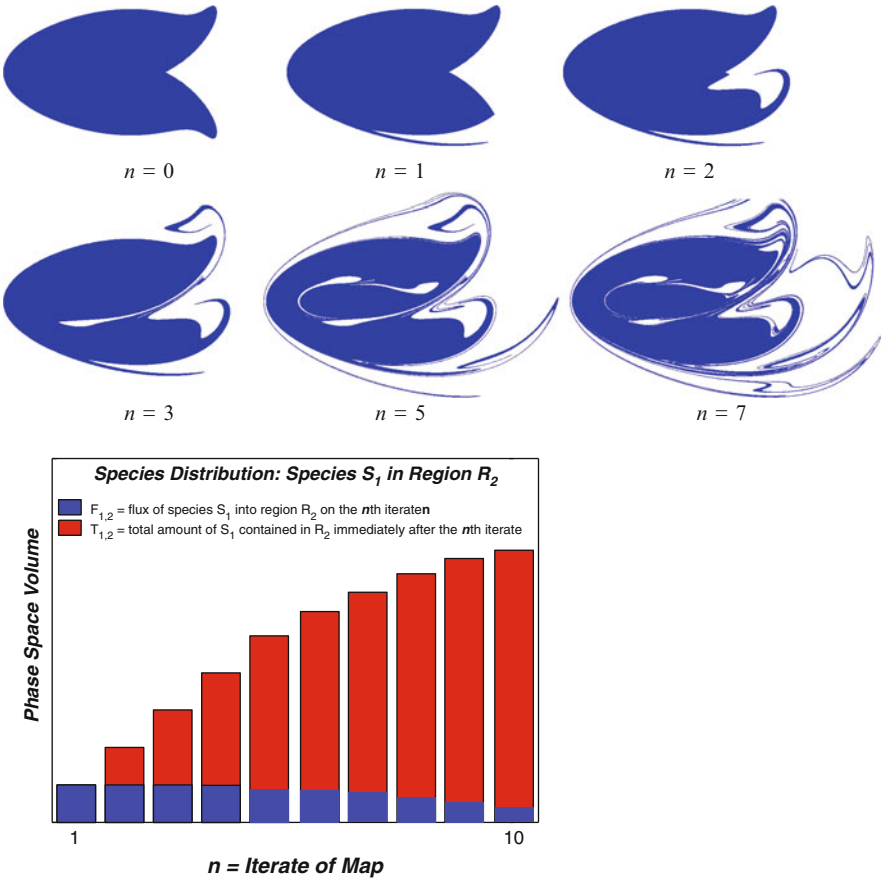

Fig. 7.9 *A multilobe, self-intersecting turnstile*. (**a**) The stable and unstable manifolds of the unstable fixed point p intersect in such a way that there are three pips between q and $f^{-1}(q)$, but our naively defined turnstile "lobes" have a non-empty intersection $I = int\left(L_{1,2}(1) \bigcap L_{2,1}(1)\right) \neq .$ (**b**) When we redefine the turnstile lobes such that $\tilde{L}_{1,2}(1) \equiv L_{1,2}(1) - I$ and $\tilde{L}_{2,1}(1) \equiv L_{2,1}(1) - I$, the result is a *multilobe, self-intersecting turnstile* consisting of a sequence of six regions; three defining $\tilde{L}_{1,2}(1)$ and three others defining $\tilde{L}_{2,1}(1)$

regions are considered. Notice that after only a few iterates of the map, a high degree of filamentation is seen.

7.2.3 *Finite-Time Analogs of Invariant Manifolds*

The method described above works for systems described by autonomous or time-periodic flow fields. However, time-chaotic flow fields do not, in general,

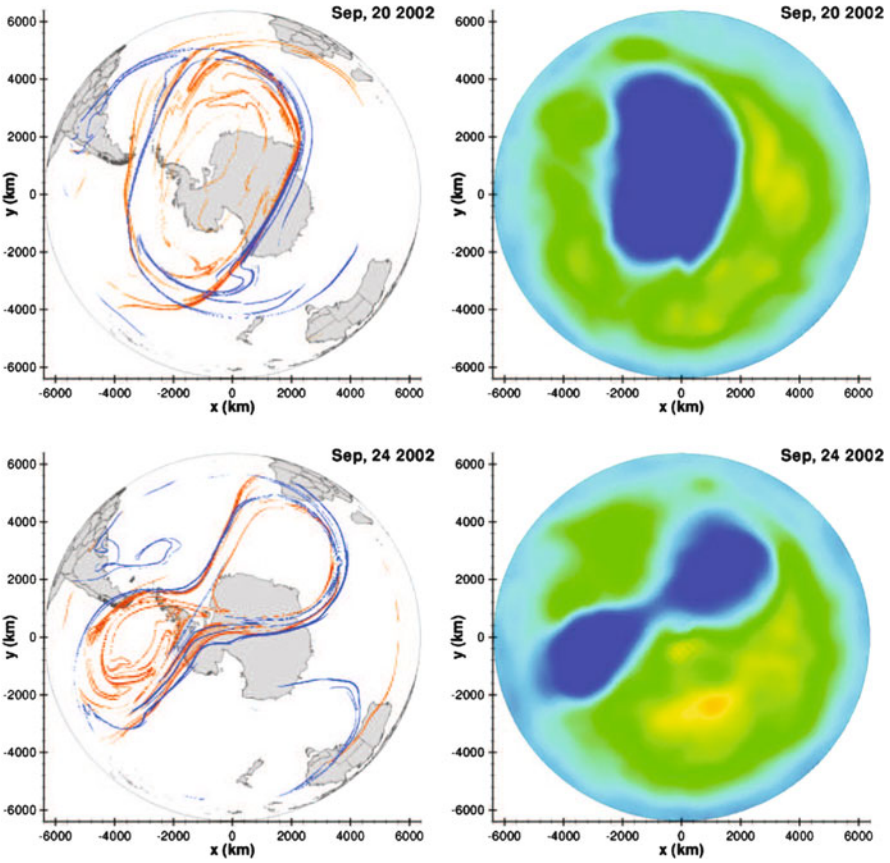

Fig. 7.10 (*Upper panels*) The two regions at time $n = 0$ define the two species (*upper left*); S_1 is dark, S_2 is white. Several iterates of S_1 are shown, namely $n = 1, 2, 3, 5, 7$. (*Lower panel*) The flux and total transport of species S_1 into region R_2 are shown as a function of iterate. The (vertical) phase space scale is relative. The example shown here comes from celestial mechanics, specifically the planar circular restricted three-body problem. The 2D state space here is from a Poincaré surface-of-section restricted to a constant energy surface

have fixed points or periodic orbits. In this case, co-dimension one surfaces of high hyperbolicity play a role analogous to the stable and unstable manifolds of NHIMs, delineating the boundaries between dynamically distinct regions and identifying lanes of transport. Computations of these surfaces have proven valuable for the analysis of geophysical transport (see Fig. 7.11). These finite-time hyperbolic manifolds, or Lagrangian coherent structures (LCS) [60, 81], can be obtained from ridges in the time-varying finite-time Lyapunov exponent (FTLE) field [28, 44, 81, 82].

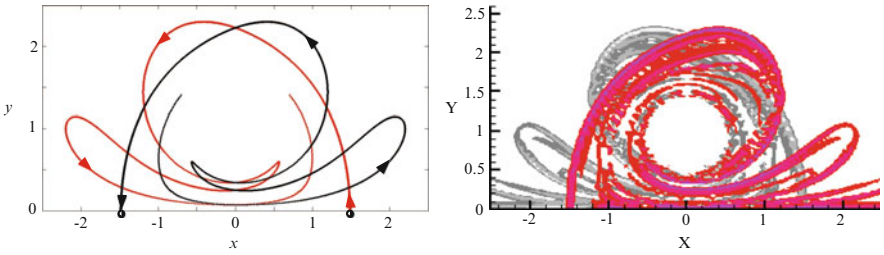

Fig. 7.11 (*Left*) Atmospheric LCS for days surrounding the Antarctic polar vortex splitting event of September 2002 (based on NCEP/NCAR reanalysis data using $T = \pm 8$ days); attracting (repelling) curves, analogous to unstable (stable) manifolds of saddle points, are shown in *blue (red)*. Before and after the splitting event in late September, we see an isolated blob of air, bounded by LCS curves, slowly rotating over Antarctica. The vortex pinches off, sending the northwestern part of the ozone hole off into the midlatitudes while the southwestern portion goes back to its regular position over Antarctica. Note the formation of lobes at the edges where chaotic stirring occurs across the LCS. (*Right*) The corresponding daily ozone concentration (based on NASA TOMS satellite data). The time-varying LCS structures that bound the ozone hole fragments provide a framework for understanding the atmospheric transport. The system can be said to have undergone a "Duffing bifurcation" given its gross similarity to a Duffing oscillator passing through a critical parameter value

7.2.3.1 FTLE Definition

For a flow field defined by data sets or a set of ordinary differential equations on a possibly non-Euclidean (i.e., curved) manifold, we define the flow map as $\phi_{t_0}^t : x(t_0) \mapsto x(t)$, which maps phase space points (or particles) from their initial location at time t_0 to their location at time t. We may be interested in FTLE and LCS on curved manifolds, e.g., where geophysical flows live, due to sphericity of the Earth or boundary topography, or to handle curved phase spaces which naturally arise in mechanics, such as $TSO(3)$ or $TSE(3)$. We define the FTLE $\sigma(x, t_0, t)$ as the norm of the differential of the flow map $\phi_{t_0}^t$ at the initial point x, i.e.,

$$\sigma(x, t_0, t) = \frac{1}{|t - t_0|} \log \left\| \mathrm{D}\phi_{t_0}^t \right\| \equiv \frac{1}{|t - t_0|} \log \left(\max_{\mathbf{y} \neq \mathbf{0}} \frac{\left\| \mathrm{D}\phi_{t_0}^t(\mathbf{y}) \right\|}{\|\mathbf{y}\|} \right), \tag{7.7}$$

where $\mathbf{y}$ is a small perturbation in the tangent space at x [28]. One can use the expression above on non-Euclidean manifolds and to derive an algorithm for computing FTLE on an unstructured mesh (for Euclidean or non-Euclidean manifolds). For Euclidean manifolds (e.g., $\mathbb{R}^n$), this expression reduces to,

$$\sigma(x, t_0, t) = \frac{1}{|t - t_0|} \log \left(\sqrt{\lambda_{\max}(C(x, t_0, t))} \right), \tag{7.8}$$

where $C(x, t_0, t) = \left(\frac{d\phi_{t_0}^t}{dx} \right)^* \left(\frac{d\phi_{t_0}^t}{dx} \right)$ is the Cauchy-Green tensor, * denotes transpose, and $\lambda_{\max}(A)$ denotes the maximum eigenvalue of matrix A

7.2.3.2 FTLE-LCS as Finite-Time Analogues of Stable and UnstableManifolds

The FTLE at x measures the maximum stretching rate over the interval $T = t - t_0$ of trajectories starting near the point x at time t_0 [35]. Ridges of high $\sigma(x, t_0, t)$ correspond to surfaces of co-dimension one, meaning they separate qualitatively different regions of flow. Repelling (attracting) LCS for $T > 0$ ($T < 0$) are the *time-dependent generalizations of stable (unstable) manifolds* of NHIMs. Consider the model of fluid particles in the presence of an oscillating vortex pair [83], a flow induced by two counter rotating vortex pairs of equal circulation. This is a time-periodic system, where the stroboscopic map has two saddle-type fixed points whose manifolds can be computed. We can compare these manifolds with the FTLE field.

Even in the absence of, e.g., saddle fixed points, one can use the analogy of repelling and attracting LCS as stable and unstable manifolds to discern transport structure. Transport occurs via alleyways bounded by undulating repelling and attracting LCS surfaces. An aperiodic version of heteroclinic-tangle chaos occurs where these surfaces intersect (see Fig. 7.11). One can consider finite-time aperiodic versions of lobe dynamics as governing phase space transport where heteroclinic or homoclinic-like intersections occur. A rich time-dependent interplay of structures is observed in geophysical systems [37], such as the atmosphere, oceans, and lakes, but much is still unknown.

For example, consider a homoclinic tangle observed in a hurricane as in Fig. 7.12. The storm shown is Andrea, the first storm of the 2007 hurricane season. Notice the outline of the U.S. east coast. One can tease out the geometry of entrainment of an air mass into a hurricane's vortex (and detrainment out of the vortex). A separatrix, defining the boundary of the storm's vortex, is first determined from intersections of the repelling and attracting curves (to give an "inside" and "outside"). Parcels are labeled according to lobe dynamics theory, and as time progresses, the parcels perform lobe dynamics predicts. The green parcel, currently outside, should get entrained. Right next to it, but on the other side of a repelling curve, is a light pink region which should remain outside. Also, the dark magenta region inside will get detrained out of the vortex. The theory of homoclinic tangles was originally developed for time-independent or time-periodic systems, but seems to work even in this turbulent, aperiodic setting. This is remarkable, since there is no actual saddle-type fixed point with which to associated the "stable" and "unstable" manifolds. Rather, a distinguished repelling and attracting LCS intersection point behaves as a saddle point for the purposes of the theory.

7.2.4 Set-Oriented Approach, Alternate FTLE Definition, and CoherentSets

A complementary method for computing transport uses almost-invariant sets (AIS), strongly connected (almost-invariant) regions computed via an operator-theoretic

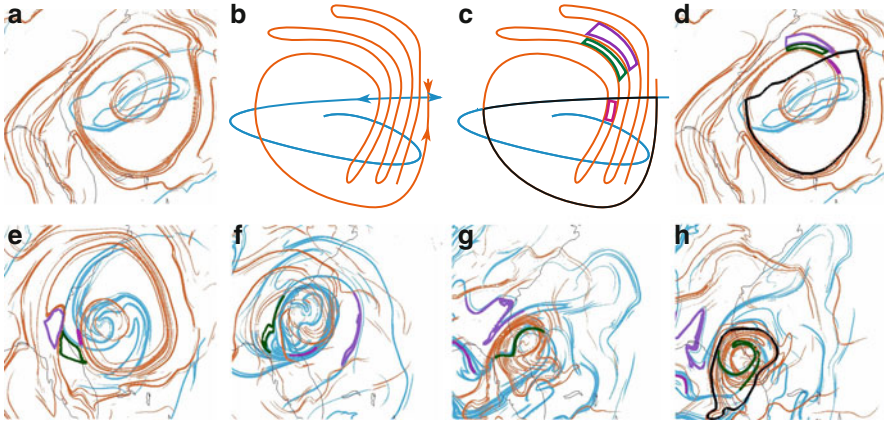

Fig. 7.12 Atmospheric LCS off the coast of Florida (coastal outline shown) for May 2007 computed from NOAA's NAM archive data at the 900 mb pressure level. The trellis-like intersections of repelling [*orange*] and attracting [*cyan*] curves (**a**) form an object which behaves like a homoclinic tangle (shown in schematic, (**b**)). Coloring according to the dictates of lobe dynamics (schematic, (**c**) and actual, (**d**)), the *green* tile will get entrained, while the *purple and magenta* tiles will remain, or be removed, outside the main vortex defined by the moving separatrix (highlighted in *black*). As time progresses ((**e**) through (**h**)), we see that the theory's predictions hold, a consequence of the geometry of intersections. For reference, the separatrix boundary of the storm's vortex is shown again in the final panel, (**h**)

approach [18, 24, 56]. AISs are defined for autonomous and time-periodic systems, but for more general time dependence, we consider coherent sets, which generalize AISs [84]. In a partition of the phase space into subsets, LCS play the role of separatrices, but coherent sets play the role of the subsets they separate; coherent sets and LCS are therefore dual structures.

7.2.4.1 Perron-Frobenius Operator and Box Discretization of Phase Space

Let μ denote the Lebesgue measure on $M \subset \mathbb{R}^n$ and $\phi(x)_{t_0}^t : M \times \mathbb{R} \times \mathbb{R} \to M$ be a flow map on M from time t_0 to t. Let B be a measurable set and $f \in L^1(M)$ be a probability density function, $L^1(M)$ being the space of Lebesgue measurable functions on M. The unique operator $\mathbb{P}_{t_0}^t : L^1(M) \mapsto L^1(M)$ defined by

$$\int_B \mathbb{P}_{t_0}^t f d\mu = \int_{(\phi_{t_0}^t)^{-1}(B)} f d\mu \tag{7.9}$$

is called the Perron-Frobenius operator for the flow $\phi_{t_0}^t$, [85]. Equation (7.10), which holds for all μ measurable sets, follows from the Radon–Nykodym theorem.

In practice it is usually necessary to numerically approximate the operator $\mathbb{P}_{t_0}^t$. This is done by discretizing the domain, M, into a finite number of sets, say

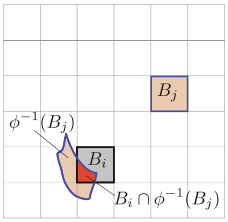

Fig. 7.13 Box-discretization method to calculate P. Box B_j at the final time t is mapped (backwards) to $\phi^{-1}(B_j)$ at the initial time t_0. The value of the entry P_{ij} is the fraction of box B_i that is mapped into box B_j by ϕ. Note that $\sum_{j=1}^{n} P_{ij} = 1$

$\{B_1, B_2, \ldots, B_n\}$ which is essentially a grid of boxes, the disjoint union of which is M, i.e., $M = \bigcup_i B_i$. A projection $\pi : L^1(M) \mapsto \text{span}\{B_1, \ldots, B_n\}$ defined by $\pi f = \sum_{i=1}^{n} c_i \mathbb{X}_{B_i}$, where $\mathbb{X}_{B_i}$ is the characteristic function of the set B_i and $c_i = \frac{\int f d\mu}{\mu(B_i)}$ gives a finite dimensional approximation of f. Since f is a probability density function, $c_i = \frac{1}{\mu(B_i)}$. Similarly, $\mathbb{P}^t_{t_0} f$ is projected on $\text{span}\{B_1, \ldots, B_n\}$. The operator $P^t_{t_0} : \pi f \mapsto \pi \mathbb{P}^t_{t_0} f$ is a linear operator between finite dimensional vector spaces. Further taking the box measures $\mu(B_i) = \mu(B_j)$ for all $i, j \in \{1, \ldots, n\}$, $P^t_{t_0}$ becomes a stochastic transition matrix. The entries of the matrix P are determined by a Monte-Carlo simulation [17, 86]. Each box in the domain contains a fixed number of points (initial conditions) which are integrated from a time t_0 to t. The final position of the points gives the matrix P as,

$$(P^t_{t_0})_{ij} = \frac{\mu(B_i \cap (\phi^t_{t_0})^{-1}(B_j))}{\mu(B_j)}. \tag{7.10}$$

See Fig. 7.13 for a geometric interpretation. A time-reversible operator P is required to apply the above definition for flows in forward time [86]. This is achieved by creating a reversible Markov operator P_r given by

$$(P^t_{t_0})_r = \frac{(P^t_{t_0}) + \overline{(P^t_{t_0})}}{2} \tag{7.11}$$

where $\overline{P}$ is the time reversed analogue of P. Its elements are given by

$$(\overline{P^t_{t_0}})_{ij} = \frac{u_{1j}(P^t_{t_0})_{ji}}{u_{1i}} \tag{7.12}$$

where u_{1j} and u_{1i} are components of the first left eigenvector, u_1, of $P^t_{t_0}$. For a volume preserving flow in which the domain is uniformly discretized, $\overline{P^t_{t_0}} = (P^t_{t_0})^*$, the transpose of $P^t_{t_0}$. Henceforth for convenience the time reversible operator $(P^t_{t_0})_r$ is referred to simply as P unless there is an ambiguity about the initial and final times t_0 and t. The Markov operator P has the semigroup property of $P^t_{t_0} = P^s_{t_0} P^t_s$, where $s \in (t_0, t)$.

7.2.4.2 Almost-Invariant Sets

A probability density function f is invariant under the flow if and only if f is a fixed point of P, i.e., $f = Pf$ [85]. The associated measure μ_f is called the invariant measure of the flow [24]. If the flow $\phi_{t_0}^t$ is volume preserving then the highest eigenvalue of P is always 1. This means that the entire domain M maps back onto itself. We make the additional observation that from (7.10), it follows that if $f > 0$, then $Pf > 0$. A set $B \in \mathbb{B}$ is considered *almost-invariant* over the interval $[t_0, t]$ if

$$\rho_{\mu_f}(B) = \frac{\mu_f(B \cap (\phi_{t_0}^t)^{-1}(B))}{\mu_f(B)} \approx 1. \tag{7.13}$$

One can in theory construct an optimization problem to maximize the value of ρ_{μ_f} over all possible combinations of sets $B \in \mathbb{B}$. But this problem is combinatorially hard to solve even for simple flows. Therefore heuristic methods advanced in [17, 86] are adopted to identify maximally AISs. It was shown in [17, 86, 87] that the left eigenvector corresponding to the second eigenvalue gives the "second most" almost-invariant sets, the invariance being higher if the second eigenvalue of P is closer to 1, with

$$\rho_{\mu_f}(B) = \frac{\sum_{i,j} u_j P_{ji}}{\sum_j u_j}. \tag{7.14}$$

For computational reasons we will make use of the singular vectors of P_r instead of the eigenvectors as was done in [88]. This is because the singular value decomposition is well conditioned and less sensitive to small changes in the matrix P_r introduced due to the discretization. It should be noted that P_r is self-adjoint and normal since $P_r = \frac{1}{2}(P + P^*) = P_r^*$ and $P_r^* P_r = P_r P_r^*$. From the spectral theorem, P_r has a unitary eigen-decomposition and the left eigenvectors of P_r are given by

$$P_r = Y^* \Lambda Y \tag{7.15}$$

with $Y^* = Y^{-1}$. P_r also has a unique singular value decomposition given by $P_r = U \Sigma V^* = P_r^* = V \Sigma U^*$ and $Y = U$, i.e., the left singular vectors and left eigenvectors of P_r are the same.

We note that the singular vectors U^* form an orthonormal basis for P_r. The first singular vector U_1^*, associated with the singular value 1, is the stationary distribution and is positive. Since $U_2^* \perp U_1^*$ we can infer that U_2^* has both positive and negative parts. From the definition of a Markov operator, $P_r(U_2^*)^+ > 0$ and $P_r(U_2^*)^- < 0$. So the sets on which the positive and negative parts of the second singular vector U_2^* are supported are AISs and partition the domain M into two parts. A 2-dimensional example from a periodic fluid velocity field in fluid mechanics is given Fig. 7.14, showing the vector U_2^* and the AISs.

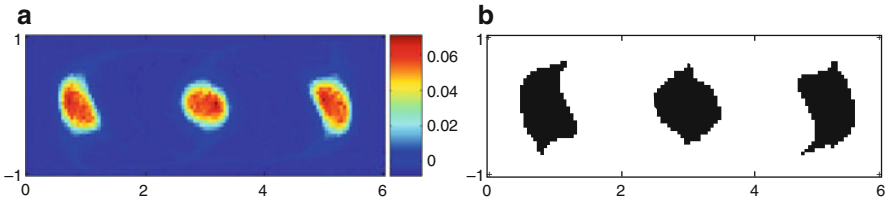

Fig. 7.14 (**a**) Second eigenvector U_2^* for the lid-driven cavity flow and (**b**) AIS obtained from positive part of the second eigenvector of P, in *black* (composed of three disconnected sets, which in this case are almost-cyclic sets of period 3). The phase space complement, in white, is also an AIS. Details in [84]

7.2.4.3 Agreement Between AIS-Based and Lobe Dynamics-Based Transport Calculations

We note that in an autonomous setting, regions defined by the lobe dynamics approach are often AISs defined using the method described above. And the rates of transport determined by the two methods—lobe areas and AIS "leakiness," respectively—are also in agreement [18], showing that, at least for autonomous or periodic systems, the geometric and probabilistic approaches are in agreement.

7.2.4.4 Coherent Sets for Systems of Arbitrary Time-Dependence

In autonomous systems an important feature of AISs is that they identify sets that mix minimally with the rest of the domain. Many time-dependent systems of practical interest may not have AISs. Moreover, systems defined by numerical data may be defined over leaky domains. In these cases, the approximation of the Perron–Frobenius operator by a stochastic transition matrix becomes difficult.

To extend the concept of AIS to time dependent systems we make the observation that AISs are also sets that stretch and deform minimally. Coherent sets—the time dependent analogues of AISs—can be defined as sets that do not mix significantly with the rest of the domain. This can be made more precise by first defining a set oriented FTLE. To make the discussion more concrete we initially define the FTLE for two dimensional systems.

Consider a uniform probability density function supported on a set B given by, $f = \frac{1}{\mu(B)}\mathbb{X}_B$, where $\mathbb{X}_B$ is characteristic function of B. The covariance matrix of f is $I_{ij} = E[(X_i - \overline{X}_i)(X_j - \overline{X}_j)]$, with $i, j = \{1, \ldots, n\}$ where $(\overline{X}_1, \ldots, \overline{X}_n)$ is the mean value of the random vector X and $E[\cdot]$ denotes the expected value. Under the action of the flow $\phi_{t_0}^t$, f is mapped to $\mathbb{P}_{t_0}^t f$ where $\mathbb{P}_{t_0}^t$ is the associated Perron-Frobenius operator. Let $I(f)$ be the covariance of f and $I(\mathbb{P}f)$ the covariance of $\mathbb{P}f$. Then the FTLE of B denoted by $\sigma_I(B, t_0, t)$ is defined as

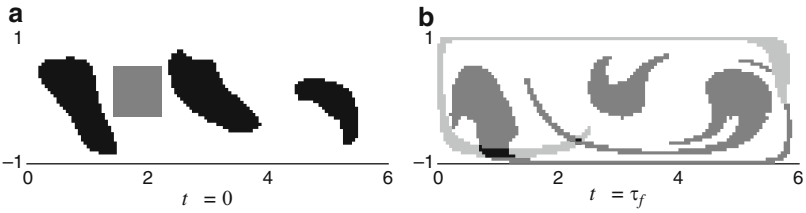

Fig. 7.15 Coherent sets (*black*) do not disperse significantly while non coherent sets (*gray*) do. In (**a**) the three coherent sets (*black*) identified by our $\sigma_I \leq 0.06$ criterion and a non-coherent set (*gray*) are shown at $t = 0$. At t equal to one period of the fluid velocity field, τ_f, the non-coherent set disperses significantly more than the coherent sets. Notice that the three coherent sets are similar to the three-components of the AIS in Fig. 7.14. In (**b**), mixing between the "particles" of the non-coherent set with those of the coherent sets is shown

$$\sigma_I(B, t_0, t) = \frac{1}{|t - t_0|} \log\left(\frac{\sqrt{\lambda_{\max}(I(\mathbb{P}f))}}{\sqrt{\lambda_{\max}(I(f))}}\right). \tag{7.16}$$

It can be shown by direct calculation that the covariance FTLE obtained from this definition and the standard FTLE definition (7.9) have the same value, i.e., $\sigma_I(B, t_0, t)$ converges to $\sigma(x, t_0, t)$ in the limit where B converges to a point x. But the σ_I definition can apply to arbitrary sets, not just points.

A set B is almost-coherent during $[t_0, t]$ if $\sigma_I(B, t_0, t) \approx 0$. This definition of coherence captures the essential feature of a coherent set: it does not mix or spread significantly in the domain. This definition also can identify non-mixing translating sets. Coherent sets can be identified by setting a heuristic threshold, say $\sigma_I^{\max}$, and considering as coherent sets the regions where $\sigma_I \leq \sigma_I^{\max}$. The probabilistic techniques described in this section are illustrated by the example of a piecewise smooth periodic lid driven cavity flow, with period τ_f (described in [31]). The AISs are shown in Fig. 7.14 while the coherent sets for the same example are shown in Fig. 7.15. Notice the similarity for this case. The coherent set definition has the advantage that it is applicable to systems of arbitrary time dependence where coherent sets could be highly mobile.

As recently discovered [31], the space-time "braiding structure" of almost-cyclic coherent sets can be analyzed using topological methods [52] to predict and quantify chaotic transport in the surrounding fluid. Since these tools are based in topology, the overall space-time structure of their interactions determines the transport predictions without requiring highly-resolved dynamical information. The merging of coherent sets and topological perspectives has great potential for making quantitative predictions of global transport with limited information.

7.3 Exploiting Chaotic Transport for Efficient Movement

Natural environmental fluid flows, such as those found in the atmosphere, oceans, lakes, and ground water, exhibit complex dynamics. This complexity is actually a boon for transporting mobile quasi-passive agents. Rich, chaotic dynamical structure makes efficient movement and dispersal possible, whether the agents are biological, chemical, or engineered devices like sensor platforms and delivery vehicles that seek energy efficiency or stealth [89]. Exploiting (or preventing) this transport requires the ability to accurately and rapidly identify these complex dynamical structures and predict their future motion. Despite the flow complexity, preliminary work suggests that it is possible to discern the time-dependent transport network of dynamical structures that geometrically organize the motion over scales of interest in realistic flows, e.g., Fig. 7.11. However, substantial work remains to be done to develop the underlying dynamical systems theory, connect these mathematical structures with experimental data, and implement efficient computation and prediction of observed structures.

7.3.1 Using Invariant Manifolds and their Time-Dependent Analogs in Efficient Control Strategies

Understanding the time-varying structures governing transport is important for advanced prediction (and control) of natural processes, but can also be exploited for engineered systems; agents navigating using invariant manifolds of saddle points (or NHIMs) or LCSs (in the case of data-based, time-aperiodic flows) can easily switch from one region of interest to another with minimal energy expenditure. Consider the schematic example of Fig. 7.16, where small controls can steer an agent into subregions which travel to desired destinations, which would not be reachable by a direct route.

Compared with conventional control methods that use only local information, incorporating global information opens new possibilities. The techniques to achieve this are a critical step in a long term vision to control multiple weakly propelled agents in 3D time-chaotic flows, in which the goal is to cooperate to achieve an overall group objective such as surveillance, search and rescue, exploration, or improved data collection or flow prediction.

The use of natural lanes of transport bounded by LCS has been extensively developed in control strategies for interplanetary spacecraft trajectory design, where in this setting the LCS are the stable and unstable manifolds of NHIMs (see [47, 51] and references therein). In this application, on-board fuel is a premium, therefore the goal is to find fuel-efficient paths which lead to, for example, an orbit around a moon of Jupiter. Although the flow field for a spacecraft lives in a six dimensional phase space—three dimensions of position and three dimensions for velocity—the computation of LCS is straightforward. The LCS reveal gravitational sweet spots that provide natural gateways to destinations. Harnessing this effect to

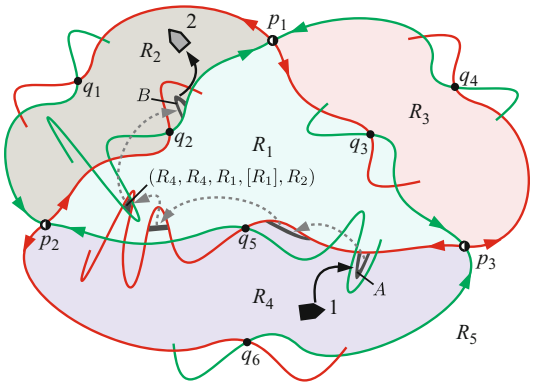

Fig. 7.16 The same regions as given in Fig. 7.6 are redrawn. A weakly propelled mobile agent initially at position 1 in region R_4 has an objective or reaching a target position 2 in region R_2, but a direct route is not possible given control limitations. The agent therefore follows a strategy of taking advantage of the existing flow structure, using a small control to enter subregion A. Under the evolution of the system (e.g., consider we have a period-T flow viewed via a time-T map), after several periods the subregion has evolved without control into B in region R_2, where a small control can take the agent to the desired position

good purpose, mission designers can plan fuel-efficient routes, ones that would not otherwise be imaginable or technically feasible.

Theoretically, LCS are expected to reveal natural fuel-free or low-fuel dynamical channels for spacecraft even in a time-dependent gravitational field [90]. This has been demonstrated numerically, where for the time-dependent flow field, the LCS are typically time-dependent surfaces [69, 70, 73, 91]. Targeting of LCS for spacecraft control has been demonstrated to reveal near optimal trajectories even in the presence of navigation or control errors [92].

Although a different setting, LCS have also been shown to approximate fuel-optimal paths for underwater gliders in coastal waters [93]. The approximately two-dimensional near-surface ocean flow is obtained from observational data and reveals rich dynamical structure with discernible features despite the time-chaotic dependence of the flow.

7.3.2 *Control Strategies for Weakly Propelled Vehicles in Complex Environmental Flows*

LCS- and AIS- or coherent set-based control can provide deployment and station-keeping where other control schemes fail, particularly for quasi-passive agents with velocities that are significantly lower than that of the environmental flow velocity [94]. We contend that vehicles (whether nano-, micro- or macro-scale) with limited power and control authority can similarly exploit a complex environmental flow for efficient transport when motion planning is informed by the evolving structures identified in the flow.

7.3.2.1 Controllability and Accessibility

Fundamental to motion planning in a time-varying flow is the question of *controllability* on a given time interval: Is there a controlled trajectory connecting *arbitrary* initial and final states? For a weakly propelled agent in a significant flow, the answer will be "no," but the system may be *accessible* in the sense that the reachable subspace is non-empty. Thus, one must ask: Does the reachable subspace contain a point or region of interest? And, if so: What control history is required? While there are well-developed notions of controllability and accessibility for nonlinear time-varying systems, constructive results (i.e., control design methods) exist only in special cases where the system exhibits an underlying structure that can be exploited to simplify analysis. For drift-free, time-invariant systems, for example, controllability can be tested using the Lie algebra rank condition [95, 96]. Having verified controllability, one may generate a time-varying input history, perhaps constructed from a library of "motion primitives," to achieve a control objective. Of course, controllability for systems with drift is more challenging.

In cases of interest to us, the primary source of drift will be the ambient flow field, which will break any natural symmetries that might otherwise be exploited in controllability analysis. Moreover, because environmental drift will *dominate* the dynamics of a weakly propelled agent, the dynamics will be accessible, at best, with severe implications for motion planning. Some key theoretical questions arise. What is the appropriate notion of accessibility for a weakly propelled agent in a time-varying flow that is characterized by LCS or coherent sets and how does one assess this property? Can structural properties of LCS or coherent sets be exploited to simplify the computations required to evaluate accessibility and characterize the reachable subspace?

To answer these questions, one must cast the control objective (e.g., waypoint following, area or volume coverage, etc.) in terms of the reachable subspaces for individual agents, given the flow field structure. Accessibility will be tied to predictions of the time-varying flow and dynamical structures. One must therefore consider a tradeoff between potential benefits of a long planning horizon (e.g., transport economy) and uncertainty in the model prediction.

7.3.2.2 Optimal Locomotion in Flows

Having developed machinery to evaluate accessibility and characterize the reachable subspace, one may then consider motion planning. What is the most energy-efficient path from point A to point B? What path from point A to point B maximizes the reachable subspace in the terminal phase? Given an initial distribution of weakly propelled agents, what collection of paths maximizes information gain (concerning the flow itself, some airborne constituent, some object on the ground, or any other measurable quantity of interest)? The first of these questions

has been considered in some detail for kinematic vehicle models in planar, constant-relative-speed motion where the flow field is steady and uniform. For a kinematic particle model with turn rate constraints, for example, there are geometric path planning strategies that concatenate maximum effort and singular arcs to create minimum-time paths [97, 98]. Turn acceleration constraints are considered in [99]. While these methods exploit (or mitigate) the effects of the ambient flow, the vehicles are "strongly propelled" and the "structure" of the flow is trivial, such as a cellular flow. We ask whether, in the more challenging conditions of a time-varying, flow field, the structural properties of LCS or coherent sets can be exploited to obtain simple, effective motion planning and control strategies for weakly propelled vehicles.

7.3.2.3 A Conceptual Example

Suppose we have a fully actuated mobile agent and we consider only translational dynamics. For simplicity of discussion, we assume that the agent is a point of unit mass. We can also suppose that there fixed obstacles to be avoided. This will lay a foundation for more detailed future studies considering the effects on a finite-size body or vehicle [100], with perhaps biologically-inspired shape change actuation [101] where coupling of translational with rotational motion is critical.

We desire a feedback control law to (asymptotically) drive the agent to a target point x_T in a target region, which may have dynamics of its own, e.g., a point on or near an LCS curve (Fig. 7.17b). A detection shell, a ball of radius r_{det}, is given to the agent such that the agent can respond to any feature within this shell, such as an LCS curve that its desirable to follow.

The agent's equation of motion are given by

$$\dot{x} = v_{\text{f}}(x,t) + v_{\text{r}} \qquad \text{and} \qquad \dot{v}_{\text{r}} = u. \tag{7.17}$$

Subscripts indicate that terms pertain to the flow ("f") and flow-relative motion ("r") due to control. The objective in this example is to drive an agent asymptotically to a target point x_T in a target region, which may have dynamics of its own, e.g., a point on or near an LCS curve (Fig. 7.17a). We are interested in the weakly propelled or *quasi-passive case* where $\|v_{\text{f}}\| < \|v_{\text{r}}\|$, where knowledge of the geometry of flow structures is critical to motion planning. Assigned to the agent is a detection shell, a ball of radius r_{det}, within which the agent may respond to flow features such as an LCS curve that might be followed toward the target. Inspired by recent developments in reactive collision avoidance [102], the agent might employ a feedback control law $u = F_p + F_g + F_d$ comprising an error potential force $F_p = -\nabla V(x)$, a gyroscopic steering force F_g, and a damping force F_d. Rather than plan an explicit path based on a flow field estimate, the agent may simply react to the evolving ambient flow conditions as they are encountered.

These forces can be chosen to illustrate three motion control methods that exploit an environmental flow characterized by an coherent set or LCSs:

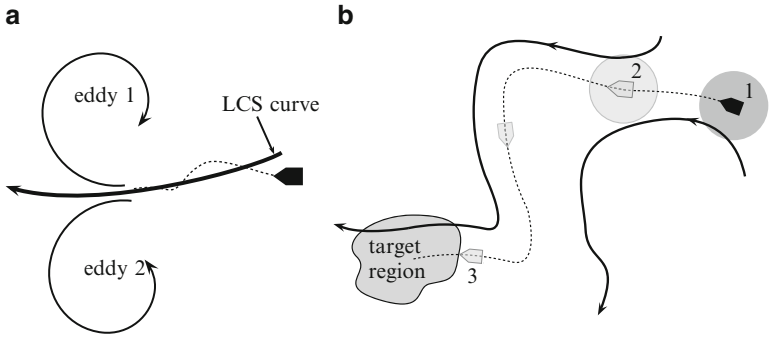

Fig. 7.17 (**a**) Straddling an LCS to easily switch between two dynamically distinct regions, for instance, two eddies. (**b**) The mobile agent follows the nearest LCS curve which is within a detection shell, shown schematically at position 1. At position 2, the agent switches to another curve, finally reaching the target region at 3. The *arrows* on the LCS curves denote the motion along them. Although not shown in this schematic, the LCS curves have dynamics of their own and will move

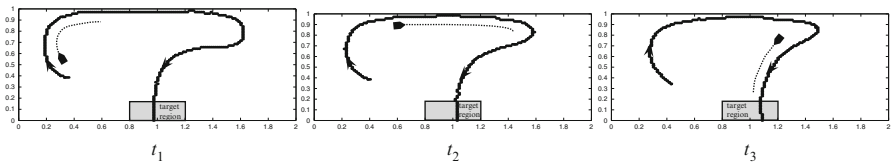

Fig. 7.18 A vehicle in a simple model of a perturbed double-gyre flow is shown at three successive times. The vehicle is using small control to take best advantage of the flow to reach the target region by following an LCS curve. The evolution of the LCS is visible as well as the short-time estimate of the vehicle's future trajectory. See [94] for details

1. Straddling an LCS for easy movement between dynamically distinct regions (AIS or coherent sets): We consider the target point x_T to be on an LCS, moving just ahead of the agent as in Fig. 7.17a. One suitable potential function is a quadratic function of the distance to the target $V(x) = \frac{1}{2}||x - x_T||^2$.
2. Following an efficient navigation pathway bounded by time-varying LCS surfaces (i.e., within an evolving coherent set) to reach a target region: We use a form of the gyroscopic force appropriate for tracking a smooth boundary [103]. If more than one LCS surface is within the agent's detection shell, the agent only reacts to the closest one, as in Fig. 7.17b.
3. Leap-frogging coherent sets to reach a destination: Passively following a coherent set while it heads in the general direction of the target region and when it veers away, selectively "jumping over" LCS to neighboring coherent sets which are headed in the right direction. This strategy views the environmental flow as an ever-changing system of "conveyor belts" moving in different directions.

The proposed strategies will quickly determine near energy-optimal trajectories: an example of strategy 2 is shown in Fig. 7.18 for a vehicle in a

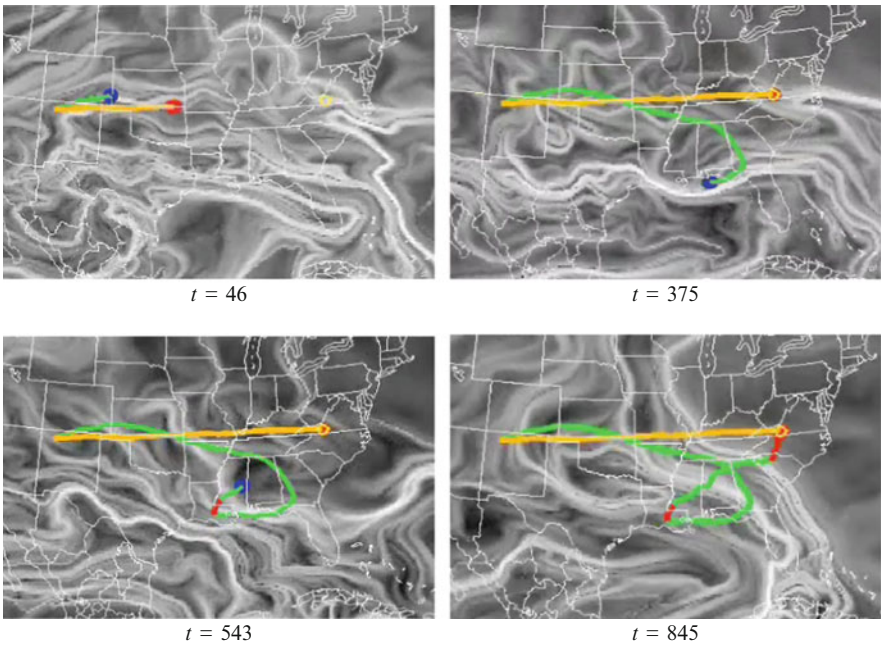

Fig. 7.19 Two aerial vehicles are experiencing atmospheric flow. Although starting and ending at the same points, a strongly propelled high speed vehicle (*straight line*) uses much more energy than a weakly propelled quasi-passive vehicle that "jumps" over LCS from one coherent set to another as needed to reach the target. The background grayscale field is the time-varying FTLE field; darker regions are more coherent, separated by ridges (LCS). The weakly propelled vehicle agent moves passively along a coherent set from $t = 0$ to around $t = 400$, when it starts heading in the wrong direction. So the vehicle applies a little control to jump over an LCS into a different coherent set heading in the right direction, with some fine control near the end to reach the target. Continental and political boundaries are shown to provide scale. Computations by Carmine Senatore

perturbed double-gyre flow [94]. A more realistic example in a high-fidelity atmospheric flow over the U.S. is shown in Fig. 7.19, displaying option 3. For any numerical optimal control algorithm, a good initial guess is vital, especially if the problem is very sensitive numerically. By having good approximate paths "built in" to the algorithms via LCS, the above control strategies will initialize any optimization strategy closer to the true optimum path. We hypothesize that this will also hold if one considers the situation where LCS can only be determined from approximate forecasts—for instance, LCS from model forecasts or assumptions on the evolution of the LCS themselves up to some finite-time horizon. One can also consider that for different motion models, generalized hyperbolic surfaces will emerge. That is, different models and controls give rise to different flow maps and thus additional generalizations of LCS which may be important for motion planning.

7.4 Conclusions

Several geometric and probabilistic methods for studying chaotic phase space transport have been developed and fruitfully applied to diverse areas from orbital mechanics to biomechanics to fluid mechanics and beyond. Increasingly, systems of interest are determined not by analytically defined model systems, but by data from experiments or large-scale simulations. We have discussed those features of phase space transport in finite-time systems which seem to be robust, considering invariant manifolds and invariant manifold-like objects, and their connection with concepts such as symbolic dynamics, almost-invariant sets, and coherent sets.

The methods outlined above provide the mathematical and computational tools necessary to identify and track dynamical structures (i.e., LCS and coherent sets) in multi-dimensional, time-dependent, complex flows in abstract phase spaces. When the focus is on fluid flows, the presence and prediction of these dynamical structures in realistic fluid flows can be exploited to achieve efficient transport of vehicles with limited power and control authority. These methods may even provide insight into the movement of weakly propelled agents in biological contexts, such as jellyfish or rays in the oceans, airborne pathogens in the atmosphere [89], or internal physiological flows.

Acknowledgements We wish to thank Carmine Senatore and Shibabrat Naik for contributing their computations. This material is based upon work supported by the National Science Foundation under Grant Nos. 0919088, 0938047 and 1100263. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

1. Koon, W.S., Lo, M.W., Marsden, J.E., Ross, S.D.: Heteroclinic connections between periodic orbits and resonance transitions in celestial mechanics. Chaos **10**, 427–469 (2000)
2. Gabern, F., Koon, W.S., Marsden, J.E., Ross, S.D.: Theory and computation of non-RRKM lifetime distributions and rates in chemical systems with three or more degrees of freedom. Physica D **211**, 391–406 (2005)
3. Gabern, F., Koon, W.S., Marsden, J.E., Ross, S.D., Yanao, T.: Application of tube dynamics to non-statistical reaction processes. Few-Body Syst. **38**, 167–172 (2006)
4. Jaffé, C., Farrelly, D., Uzer, T.: Transition state in atomic physics. Phys. Rev. A **60**, 3833–3850 (1999)
5. Jaffé, C., Farrelly, D., Uzer, T.: Transition state theory without time-reversal symmetry: chaotic ionization of the hydrogen atom. Phys. Rev. Lett. **84**, 610–613 (2000)
6. Martens, C.C., Davis, M.J., Ezra, G.S.: Local frequency analysis of chaotic motion in multidimensional systems: energy transport and bottlenecks in planar OCS. Chem. Phys. Lett. **142**, 519–528 (1987)
7. De Leon, N.: Cylindrical manifolds and reactive island kinetic theory in the time domain. J. Chem. Phys. **96**, 285–297 (1992)
8. De Leon, N., Ling, S.: Simplification of the transition state concept in reactive island theory: Application to the HCN=CNH isomerization. J. Chem. Phys. **101**, 4790–4802 (1994)

9. De Leon, N., Mehta, M.A., Topper, R.Q.: Cylindrical manifolds in phase space as mediators of chemical reaction dynamics and kinetics. I. Theory. J. Chem. Phys. **94**, 8310–8328 (1991)
10. De Leon, N., Mehta, M.A., Topper, R.Q.: Cylindrical manifolds in phase space as mediators of chemical reaction dynamics and kinetics. II. Numerical considerations and applications to models with two degrees of freedom. J. Chem. Phys. **94**, 8329–8341 (1991)
11. Shadden, S.C., Lekien, F., Marsden, J.: Definition and properties of Lagrangian coherent structures from finite-time Lyapunov exponents in two-dimensional aperiodic flows. Physica D **212**, 271–304 (2005)
12. Lekien, F., Shadden, S.C., Marsden, J.: Lagrangian coherent structures in *n*-dimensional systems. J. Math. Phys. **48** (2007)
13. Wiggins, S.: Chaotic Transport in Dynamical Systems, Interdisciplinary Applied Mathematics, 1st edn. Springer, Berlin (1993)
14. Ide, K., Small, D., Wiggins, S.: Distinguished hyperbolic trajectories in time dependent fluid flows: Analytical and computational approach for velocity fields as data sets. Nonlin. Process. Geophys. **9**, 237–263 (2002)
15. Wiggins, S.: The dynamical systems approach to Lagrangian transport in oceanic flows. Ann. Rev. Fluid Mech. **37**, 295–328 (2005)
16. Haller, G.: Lagrangian coherent structures and the rate of strain in a partition of two-dimensional turbulence. Phys. Fluids A **13**, 3368–3385 (2001)
17. Dellnitz, M., Junge, O.: On the approximation of complicated dynamical behavior. SIAM J. Numer. Anal. **36**, 491–515 (1998)
18. Dellnitz, M., Junge, O., Koon, W.S., Lekien, F., Lo, M.W., Marsden, J.E., Padberg, K., Preis, R., Ross, S.D., Thiere, B.: Transport in dynamical astronomy and multibody problems. Int. J. Bifurc. Chaos **15**, 699–727 (2005)
19. Dellnitz, M., Junge, O., Lo, M.W., Marsden, J.E., Padberg, K., Preis, R., Ross, S.D., Thiere, B.: Transport of Mars-crossing asteroids from the quasi-Hilda region. Phys. Rev. Lett. **94**, 231102 (2005)
20. Froyland, G., Dellnitz, M.: Detecting and locating near optimal almost invariant sets and cycles. SIAM J. Sci. Comput. **24**, 1507–1523 (2009)
21. Froyland, G., Lloyd, S., Santitissadeekorn, N.: Coherent sets for nonautonomous dynamical systems. Physica D **239**, 1527–1541 (2010)
22. Froyland, G., Santitissadeekorn, N., Monahan, A.: Transport in time-dependent dynamical systems: Finite time coherent sets. Chaos **20**, 043116 (2010)
23. Du Toit, P.C.: Transport and separatrices in time dependent flows. PhD thesis, California Institute of Technology (2010)
24. Dellnitz, M., Hohmann, A., Junge, O., Rumpf, M.: Exploring invariant sets and invariant measures. Chaos **7**, 221 (1997)
25. Coulliette, C., Lekien, F., Paduan, J.D., Haller, G., Marsden, J.: Optimal pollution mitigation in monterey bay based on coastal radar data and nonlinear dynamics. Environ. Sci. Technol. **41**, 6562–6572 (2007)
26. Tallapragada, P., Ross, S.D.: Particle segregation by Stokes number for small neutrally buoyant spheres in a fluid. Phys. Rev. E **78** (2008)
27. Haller, G., Sapsis, T.: Instabilities in the dynamics of neutrally buoyant particles. Phys. Fluids **20** (2008)
28. Lekien, F., Ross, S.D.: The computation of finite-time Lyapunov exponents on unstructured meshes and for non-Euclidean manifolds. Chaos **20**, 017505 (2010)
29. Tang, W., Mathur, M., Haller, G., Hahn, D.C., Ruggiero, F.H.: Lagrangian coherent structures near a subtropical jet stream. J. Atmos. Sci. **67**, 2307–2319 (2010)
30. Wilson, M., Peng, J., Dabiri, J.O., Eldredge, J.D.: Lagrangian coherent structures in low Reynolds number swimming. J. Phys. Conden. Matter **21**, 204105 (2009)
31. Stremler, M.A., Ross, S.D., Grover, P., Kumar, P.: Topological chaos and periodic braiding of almost-cyclic sets. Phys. Rev. Lett. **106**, 114101 (2011)

32. Cross, M.C., Hohenberg, P.C.: Pattern formation outside of equilibrium. Rev. Mod. Phys. **65**(3), 851 (1993)
33. Pierrehumbert, R.T.: Chaotic mixing of tracer and vorticity by modulated traveling Rossby waves. Geophys. Astrophys. Fluid Dyn. **58**, 285–319 (1991)
34. Pierrehumbert, R.T.: Large-scale horizontal mixing in planetary atmospheres. Phys. Fluids A **3**, 1250–1260 (1991)
35. Shadden, S.C., Lekien, F., Marsden, J.E.: Definition and properties of Lagrangian coherent structures: mixing and transport in two-dimensional aperiodic flows. Physica D **212**, 271–304 (2005)
36. Haynes, P.: Stratospheric dynamics. Annu. Rev. Fluid Mech. **37**, 263–293 (2005)
37. Du Toit, P.C., Marsden, J.E.: Horseshoes in hurricanes. J. Fixed Point Theory Appl. **7**(2), 351–384 (2010)
38. Dingwell, J.B., Cusumano, J.P.: Nonlinear time series analysis of normal and pathological human walking. Chaos **10**(4), 848–863 (2000)
39. Dingwell, J.B.: Lyapunov exponent. In: Akay, M. (ed.) The Wiley Encyclopedia of Biomedical Engineering. Wiley, New York (2006)
40. England, S., Granata, K.P.: The influence of gait speed on local dynamic stability of walking. Gait Posture **25**, 172–178 (2007)
41. Tanaka, M.L., Ross, S.D.: Separatrices and basins of stability from time series data: an application to biodynamics. Nonlin. Dyn. **58**(1–2), 1–21 (2009)
42. Tanaka, M.L., Nussbaum, M.A., Ross, S.D.: Evaluation of the threshold of stability for the human spine. J. Biomech. **42**(8), 1017–1022 (2009)
43. Tanaka, M.L., Ross, S.D., Nussbaum, M.: Mathematical modeling and simulation of seated stability. J. Biomech. **43**, 906–912 (2010)
44. Ross, S.D., Tanaka, M.L., Senatore, C.: Detecting dynamical boundaries from kinematic data in biomechanics. Chaos **20**, 017507 (2010)
45. Tanaka, M.L., Ross, S.D.: Using topological equivalence to discover stable control parameters in biodynamic systems. Comput. Meth. Biomech. Biomed. Eng. (2012). doi:10.1080/10255842.2011.565413
46. Hénon, M., Heiles, C.: The applicability of the third integral of motion: some numerical experiments. Astron. J. **69**, 73–79 (1964)
47. Koon, W.S., Lo, M.W., Marsden, J.E., Ross, S.D.: Systems, the Three-Body Problem and Space Mission Design. Marsden Books, ISBN 978-0-615-24095-4
48. Ross, S.D.: The interplanetary transport network. Am. Sci. **94**, 230–237 (2006)
49. Ross, S.D., Scheeres, D.J.: Multiple gravity assists, capture, and escape in the restricted three-body problem. SIAM J. Appl. Dyn. Syst. **6**(3), 576–596 (2007)
50. Grover, P., Ross, S.D.: Designing trajectories in a planet-moon environment using the controlled Keplerian map. J. Guid. Contr. Dyn. **32**, 436–443 (2009)
51. Gawlik, E.S., Marsden, J.E., Du Toit, P.C., Campagnola, S.: Lagrangian coherent structures in the planar elliptic restricted three-body problem. Celestial Mech. Dyn. Astron. **103**, 227–249 (2009)
52. Boyland, P.L., Aref, H., Stremler, M.A.: Topological fluid mechanics of stirring. J. Fluid Mech. **403**, 227–304 (2000)
53. Thiffeault, J.-L., Finn, M.D.: Topology, braids and mixing in fluids. Phil. Trans. R. Soc. Lond. A **364**(1849), 3251–3266 (2006)
54. Gouillart, E., Thiffeault, J., Finn, M.: Topological mixing with ghost rods. Phys. Rev. E **73**(3) (2006)
55. Stremler, M.A., Chen, J.: Generating topological chaos in lid-driven cavity flow. Phys. Fluids **19**, 103602 (2007)
56. Dellnitz, M., Junge, O.: On the approximation of complicated dynamical behavior. SIAM J. Numer. Anal. **36**, 491–515 (1999)
57. Jaffé, C., Ross, S.D., Lo, M.W., Marsden, J.E., Farrelly, D., Uzer, T.: Theory of asteroid escape rates. Phys. Rev. Lett. **89**, 011101 (2002)

58. Rom-Kedar, V., Wiggins, S.: Transport in two-dimensional maps. Arch. Rat. Mech. Anal. **109**, 239–298 (1990)
59. Rom-Kedar, V., Wiggins, S.: Transport in two-dimensional maps: Concepts, examples, and a comparison of the theory of Rom-Kedar and Wiggins with the Markov model of MacKay, Meiss, Ott, and Percival. Physica D **51**, 248–266 (1991)
60. Haller, G., Yuan, G.: Lagrangian coherent structures and mixing in two-dimensional turbulence. Physica D **147**, 352–370 (2000)
61. Haller, G.: Finding finite-time invariant manifolds in two-dimensional velocity fields. Chaos **10**, 99–108 (2000)
62. Haller, G.: Distinguished material surfaces and coherent structures in 3D fluid flows. Physica D **149**, 248–277 (2001)
63. Haller, G.: Lagrangian structures and the rate of strain in a partition of two-dimensional turbulence. Phys. Fluids **13**, 3365–3385 (2001)
64. Haller, G.: Lagrangian coherent structures from approximate velocity data. Phys. Fluids **14**, 1851–1861 (2002)
65. Lekien, F., Shadden, S.C., Marsden, J.E.: Lagrangian coherent structures in *n*-dimensional systems. J. Math. Phys. **48**, 065404 (2007)
66. Bowman, K.P., Pan, L.L., Campos, T., Gao, R.: Observations of fine-scale transport structure in the upper troposphere from the High-performance Instrumented Airborne Platform for Environmental Research. J. Geophys. Res. **112**, D18111 (2007)
67. Pan, L.L., Bowman, K.P., Atlas, E., Wofsy, S.C., Zhang, F., Bresch, J.F., Ridley, B.A., Pittman, J.V., Homeyer, C., Romashkin, P., Cooper, W.A.: The stratosphere-troposphere analyses of regional transport 2008 (START08) experiment. Bull. Am. Meteorol. Soc. **91**, D23102 (2010)
68. Koon, W.S., Lo, M.W., Marsden, J.E., Ross, S.D.: Resonance and capture of Jupiter comets. Celestial Mech. Dyn. Astron. **81**, 27–38 (2001)
69. Koon, W.S., Lo, M.W., Marsden, J.E., Ross, S.D.: Low energy transfer to the Moon. Celestial Mech. Dyn. Astron. **81**: 63–73 (2001)
70. Koon, W.S., Lo, M.W., Marsden, J.E., Ross, S.D.: Constructing a low energy transfer between Jovian moons. Contemp. Math. **292**, 129–145 (2002)
71. Koon, W.S., Marsden, J.E., Ross, S.D., Lo, M.W., Scheeres, D.J.: Geometric mechanics and the dynamics of asteroid pairs. Ann. New York Acad. Sci. **1017**, 11–38 (2004)
72. Ross, S.D.: Cylindrical manifolds and tube dynamics in the restricted three-body problem. Ph.D. thesis, California Institute of Technology (2004)
73. Marsden, J.E., Ross, S.D.: New methods in celestial mechanics and mission design. Bull. Am. Math. Soc. **43**, 43–73 (2006)
74. Newton, P.K., Ross, S.D.: Chaotic advection in the restricted four-vortex problem on a sphere. Physica D **223**, 36–53 (2006)
75. Jerg, S., Junge, O., Ross, S.D.: Optimal capture trajectories using multiple gravity assists. Commun. Nonlin. Sci. Numer. Simul. **14**(12), 4168–4175 (2009)
76. Gómez, G., Koon, W.S., Lo, M.W., Marsden, J.E., Masdemont, J., Ross, S.D.: Connecting orbits and invariant manifolds in the spatial three-body problem. Nonlinearity **17**, 1571–1606 (2004)
77. Wiggins, S.: Chaotic Transport in Dynamical Systems, vol. 2 of Interdisciplinary Applied Mathematics. Springer, Berlin (1992)
78. Meiss, J.D.: Symplectic maps, variational principles, and transport. Rev. Mod. Phys. **64**, 795–848 (1992)
79. MacKay, R.S., Meiss, J.D., Percival, I.C.: Transport in Hamiltonian systems. Physica D **13**, 55–81 (1984)
80. Schroer, C.G., Ott, E.: Targeting in Hamiltonian systems that have mixed regular/chaotic phase spaces. Chaos **7**, 512–519 (1997)
81. Tallapragada, P., Ross, S.D.: Particle segregation by Stokes number for small neutrally buoyant spheres in a fluid. Phys. Rev. E **78**, 036308 (2008)

82. Senatore, C., Ross, S.D.: Detection and characterization of transport barriers in complex flows via ridge extraction of the finite time Lyapunov exponent field. Int. J. Numer. Methods Eng. **86**, 1163–1174 (2011)
83. Rom-Kedar, V., Leonard, A., Wiggins, S.: An analytical study of transport, mixing and chaos in an unsteady vortical flow. J. Fluid Mech. **214**, 347–394 (1990)
84. Tallapragada, P., Ross, S.D.: A set oriented definition of the finite-time Lyapunov exponents and coherent sets (2012) (under review)
85. Lasota, A., Mackey, M.C.: Chaos, Fractals and Noise. Stochastic Aspects of Dynamics, 2nd edn. Springer, Berlin (1994)
86. Froyland, G., Dellnitz, M.: Detecting and locating near-optimal almost-invariant sets and cycles. SIAM J. Sci. Comput. **24**, 1839–1863 (2003)
87. Dellnitz, M., Junge, O.: Set oriented numerical methods for dynamical systems. In: Handbook of dynamical systems, vol. 2, pp. 221–264. North-Holland, Amsterdam (2002)
88. Fritzsche, D., Mehrmann, V.: An SVD approach to identifying metastable sets of Markov chains. In: Technical Report, Institut fur Mathematik, Technishe Universitat Berlin, 2006
89. Tallapragada, P., Ross, S.D., Schmale, D.G.: Lagrangian coherent structures are associated with fluctuations in airborne microbial populations. Chaos **21**, 033122 (2011)
90. Fenichel, N.: Persistence and smoothness of invariant manifolds for flows. Indiana Univ. Math. J. **21**, 193–226 (1971)
91. Yamato, H., Spencer, D.B.: Transit-orbit search for planar restricted three-body problems with perturbations. J. Guid. Contr. Dyn. **27**, 1035–1045 (2004)
92. Serban, R., Koon, W., Lo, M., Marsden, J., Petzold, L., Ross, S.D., Wilson, R.: Halo orbit mission correction maneuvers using optimal control. Automatica **38**, 571–583 (2002)
93. Inanc, T., Shadden, S.C., Marsden, J.E.: Optimal trajectory generation in ocean flows. In: Proceedings of 2005 American Control Conference, pp. 674–679, Portland, OR, 2005
94. Senatore, C., Ross, S.D.: Fuel-efficient navigation in complex flows. In: Proceedings of 2008 American Control Conference, pp. 1244–1248 (2008)
95. Nijmeiejer, H., van der Schaft, A.J.: Nonlinear Dynamical Control Systems. Springer, Berlin (1990)
96. Bloch, A.M.: Nonholomomic mechanics and control. Applied Mathematical Sciences Series. Springer, Berlin (2003)
97. Techy, L., Woolsey, C.A.: Minimum-time path planning for unmanned aerial vehicles in steady uniform winds. J. Guid. Contr. Dyn. **32**(6), 1736–1746 (2009)
98. McGee, T.G., Hedrick, J.K.: Optimal path planning with a kinematic airplane model. J. Guid. Contr. Dyn. **30**(2), 629–633 (2007)
99. Techy, L., Woolsey, C.A., Morgansen, K.A.: Minimum-time planar path planning for flight vehicles in wind with turn rate and acceleration bounds. In: IEEE International Conference on Robotics and Automation, pp. 3240–3245. Anchorage, AK (2010)
100. Cartwright, J.H.E., Magnasco, M.O., Piro, O., Tuval, I.: Bailout embeddings and neutrally buoyant particles in three-dimensional flows. Phys. Rev. Lett. **89**, 264501 (2002)
101. Ross, S.D.: Optimal flapping strokes for self-propulsion in a perfect fluid. In: Proceedings of 2006 American Control Conference, pp. 4118–4122 (2006)
102. Chang, D.E., Marsden, J.E.: Gyroscopic forces and collision avoidance with convex obstacles. In: Kang, W., Xiao, M., Borges, C. (eds.) New Trends in Nonlinear Dynamics and Control, and their Applications. Lecture Notes in Control and Information Sciences, vol. 295, pp. 145–160. Springer, Berlin (2003)
103. Zhang, F., Justh, E., Krishnaprasad, P.S.: Boundary following using gyroscopic control. In: Proceedings of 43rd IEEE Conference on Decisions and Control, pp. 5204–5209 (2004)

Part 5
Chaos Theory in Communication and Cryptography

Chapter 8
Chaos Applications in Digital Watermarking

Congxu Zhu and Kehui Sun

8.1 Chaos-Based Robust Image Watermarking

Digital watermarking can provide an effective scheme for copyright protection and content authorization of digital products. In this section, we would limit our discussion to digital watermarks serving the purpose of copyright protection. Digital watermarks of this type should be perceptually and statistically invisible, and the watermark is robust against typical signal processing operations. Hence, this kind of watermarking is called robust image watermarking.

The watermarking techniques proposed so far can be classified in two broad categories [3]: (a) Methods that embed the watermark by directly modifying the intensity of certain pixels, which is called spatial domain digital watermarking. (b) Methods that act upon selected coefficients of a properly chosen transform domain (DCT domain, DFT domain, etc.), which are collectively referred to as transform domain digital watermarking. The former can embed a large number of bits without incurring noticeable visual artifacts; whereas, the latter has been shown to be quite robust against lossy compression, filtering, and noise pollution, etc.

C. Zhu
School of Information Science and Engineering, Central South University, Changsha, 410083, China
e-mail: zhucx@csu.edu.cn

K. Sun (✉)
School of Physics Science and Technology, Central South University, Changsha, 410083, China
e-mail: kehui@csu.edu.cn

S. Banerjee et al. (eds.), *Applications of Chaos and Nonlinear Dynamics in Science and Engineering - Vol. 2*, Understanding Complex Systems,
DOI 10.1007/978-3-642-29329-0_8,

8.1.1 Chaos-Based Image Spatial Domain Robust Digital Watermarking Algorithm

Most spatial domain schemes embed the watermark in the LSB (least significant bit) of image pixels, but the robustness against attacks is weak, and watermark can be detected easily, many new schemes for LSB algorithm are proposed to improve the robustness, but they are not secure enough. In [4], Hash function was employed to improve the security of the water marking algorithm, in [5], a spatial domain robust digital watermarking approach that will not degrade the quality of host image was proposed, but a mapping table is needed in recording the embedding position and in watermark extraction, the use of mapping table increased the complexity of the algorithm, it is not a blind extraction algorithm in a sense.

In this subsection, a robust image spatial domain blind watermarking algorithm is presented. In the scheme, two-dimensional Arnold cat map is employed to shuffle the embedding position, therefore, watermark signals spread in all regions of the host image chaotically, which ensures the security of watermarking algorithm. To tradeoff between imperceptibility and robustness, watermark bits are used to modify the 4th, 5th, 6th or 7th bits of corresponding shuffled pixels in host image. Original image and mapping table are not needed in watermark extraction, so it is a blind watermark scheme.

8.1.1.1 Chaotic Maps and Embedding Position of Watermark

Chaotic signals, which are complex and impossible to predict over long time, can be generated by simple dynamic systems such as Logistic map and the skew tent chaotic map. A large number of uncorrelated, random-like, yet deterministic chaotic signals can be generated with small perturbation of parameters. Keeping the chaotic parameters and initial condition as the secret key, the chaotic signal can be reproduced easily.

In order to shuffle the embedding position of the host image, two dimensional Arnold cat map is employed in our scheme, which is described by [6]:

$$\begin{cases} x_{n+1} = (x_n + y_n) \mod 1 \\ y_{n+1} = (x_n + 2y_n) \mod 1 \end{cases} \tag{8.1}$$

where notation "x mod1" denotes the fractional part of a real number x by adding or subtracting an appropriate integer. Therefore, (x_n, y_n) is confined in a unit square of $[0, 1] \times [0, 1]$, write formula (8.1) in matrix form, we obtain:

$$\begin{pmatrix} x_{n+1} \\ y_{n+1} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} x_n \\ y_n \end{pmatrix} = C \begin{pmatrix} x_n \\ y_n \end{pmatrix} \mod 1 \tag{8.2}$$

A unit square is first stretched by linear transformation and then folded by modulo operation, so the cat map is area preserving, and the determinant of its linear transformation matrix |A| is equal to 1. The map is known to be chaotic, and it is one to one map. Each point of the unit square is uniquely mapped onto another point in the unit square.

The cat map above can be extended as follows. Firstly, the phase space is generalized to [0, 1, 2,. . ., $N-1$] × [0, 1, 2,. . ., $N-1$], i.e., only positive integers from 0 to $N-1$ are taken, then (8.2) is generalized to two dimensional invertible chaotic map:

$$\begin{pmatrix} x_{n+1} \\ y_{n+1} \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x_n \\ y_n \end{pmatrix} = C \begin{pmatrix} x_n \\ y_n \end{pmatrix} \bmod N, \tag{8.3}$$

where a, b, c and d are positive integers, and $|A| = ad - bc = 1$, therefore, only three among four parameters of a, b, c and d are independent under this condition. The map (8.3) is also known to be chaotic and is one to one map, each point in the square of $[0, N-1] \times [0, N-1]$ is uniquely mapped onto another point in the square of [0, $N-1$] × [0, $N-1$].

By using the generalized cat map (8.3), the embedding position of pixels from watermark image to host image can be obtained, i.e., the coordinate (i, j) of watermark pixel is served as the initial value, three independent parameters as well as the iteration times n as the secret key, after n rounds of iterations, the iterating result (x_n, y_n) will be served as the embedding position of the watermark pixel (i, j). When the iteration times n are big enough, arbitrary adjacent two watermark pixels will separate apart largely in the host image, different watermark pixels will get different embedding positions, so the embedded watermark pixels will spread in host image randomly.

To determine the bit of host image pixels in watermark embedding, another chaotic map is used in the scheme. A commonly used one is the Logistic map, which is described by

$$z_{n+1} = \mu z_n (1 - z_n), \tag{8.4}$$

Where $z_n \in (0, 1)$, $\mu \in (0, 4)$]. When $\mu > 3.5699456$, the sequence iterated with initial value z_0 is chaotic, different sequences will be generated with different initial values. Because the sequence is normally distributed in the interval of (0, 1) and is non-periodic, the interval of (0, 1) can be divided into several subintervals which correspond to different pixel bits for watermark embedding.

8.1.1.2 Watermark Embedding and Extraction

(1) Watermark embedding

Let the binary watermark of size $M_1 \times M_2$ be denoted as $W = \{w(i, j), 1 \le i \le M_1, 1 \le j \le M_2\}$, the original host image of size $N_1 \times N_2$ be denoted as $F = \{f(x, y),$

$1 \le x \le N_1$, $1 \le y \le N_2\}$, where (i, j) and (x, y) represent the pixel coordinates of binary watermark image and original host image, respectively, $w(i, j) = \{0, 1\}$, $f(x, y) = \{0, 1, \ldots, 2L - 1\}$, where L denotes the binary bits of gray level image pixels.

For convenience, we assume $M_1 = M_2 = M$, $N_1 = N_2 = N$. Considering the security of watermark embedding (avoid unauthorized extraction), watermark bits (1 bit per pixel) are embedded randomly to the pixel bits in host image. The embedding position (x, y) is calculated by formula (8.3), that is to say, take the position (i, j) of watermark pixel as initial value, iterate n times to obtain (x, y), three independent parameters and iteration times n are kept as the private key, different watermark position (i, j) will map on different embedding position (x, y). Furthermore, no record table is needed to record the colliding positions, as Arnold cat map is one to one map. The watermark pixel is embedded to the kth bit of pixel (x, y) in host image, where $k = 4, 5, 6, 7$ is determined by the subinterval of z_n generated by Logistic map (4), so the bit position to be embedded is located by the coordinate (x, y) and k. Let the host image pixel embedded watermark be denoted as $f'(x, y)$, if $w(i, j)$ is the same as the kth bit of $f(x, y)$, then $f'(x, y) = f(x, y)$, i.e., the pixel value is kept unchanged; otherwise, the kth bit of $f(x, y)$ is substituted by $w(i, j)$.

Watermark embedding algorithm can be described as follows.

Step 1: Designate three independent parameters of generalized Arnold cat map, iteration times n as well as initial value z_0 of Logistic map.

Step 2: Giving a group of coordinate value (i, j) of watermark pixel, the corresponding watermark pixel value is $w(i, j)$, let $x_0 = i$, $y_0 = j$, iterate n times by formula (8.3) to obtain (x, y).

Step 3: Perform Logistic iteration to obtain a real sequence z_n, where $z_n \in (0, 1)$, then determine k: $z_n \in (0, 0.25]$, $k = 4$; $z_n \in (0.25, 0.5]$, $k = 5$; zn $\in (0.5, 0.75]$, $k = 6$; $z_n \in (0.75, 1)$, $k = 7$. Find out the kth bit b_k from $f(x, y)$.

Step 4: If $w(i, j) = b_k$, then $f'(x, y) = f(x, y)$; otherwise $f'(x, y) = w(i, j)$.

Step 5: Giving another group of coordinate value (i, j) such that the next watermark pixel is obtained; repeat step 2 through step 4 until all watermark pixels are embedded.

(2) Watermark extraction

Watermark extraction is just the inverse process of the above embedding algorithm. In the process of extraction, one needs to know the key parameters and initial value z_0 as well as watermark length, let $w'(i, j)$ denote the extracted watermark pixel, watermark extraction algorithm can be described as follows.

Step 1: Designate three independent parameters, initial watermark position value (i, j) and iteration times n for Arnold cat map, as well as the initial value z_0 of logistic map.

Step 2: For watermark pixel $w'(i, j)$ to be extracted, let $x_0 = i$, $y_0 = j$, iterate n times by formula (8.3) to obtain (x, y).

Fig. 8.1 Invisibility and extractability test: (**a**) Original "Lena" image; (**b**) Watermark logo; (**c**) Watermarked image; (**d**) Extracted watermark with right secret keys. (**e**) Extracted watermark with wrong key z0

Step 3: Perform the same logistic iteration as the embedding algorithm to obtain z_n, $z_n \in (0, 0.25]$, $k = 4$; $z_n \in (0.25, 0.5]$, $k = 5$; $z_n \in (0.5, 0.75]$, $k = 6$; $z_n \in (0.75, 1)$, $k = 7$.

Step 4: Calculate the kth bit p_k of $f'(x, y)$, and obtain $w'(i, j) = p_k$.

Step 5: Giving another group of coordinate value (i, j) and Repeating step 2 through step 4 until all the watermark pixels $w'(i, j)$ $(i, j = 1, 2, \ldots, M)$ are extracted.

In the extraction procedure, original image and recording table are not needed except for secret key, therefore, it is a blind watermarking scheme.

8.1.1.3 Simulation Results

To demonstrate the effectiveness of the proposed algorithm, MATLAB simulations are performed by using 256×256 pixel gray level "Lena" image and 64×64 pixel binary watermark logo "Central South University (in Chinese)". In our algorithm, three independent parameters of Arnold cat map are $a = 1$, $b = 2$, $c = 3$, respectively, the number of iteration $n = 20$, μ is taken as 4, the initial value of Logistic map $z_0 = 0.2$. The watermark bits are embedded into 2th, 3th, 4th, 5th or 6th bits of the pixel (x, y) in host image randomly.

In the simulation, we first performed invisibility and extractability test of watermark. Figure 8.1 demonstrates the invisibility of watermark. Figure 8.1a, b show the original host image and watermark logo respectively. Figure 8.1c shows the watermarked image. Figure 8.1d, e show the extracted watermark logo "Central South University (in Chinese)" with right secret keys and wrong key z_0, respectively. One can see that the embedded watermark is perceptually invisible. The wrong key z_0 in watermark extraction process has only a difference of 0.00001 from the right key z_0.

As a performance measurement for image distortion, the well known peak-signal-to-noise ratio (*PSNR*) which is classified under the difference distortion metrics can be applied on the stego-images. It is defined by

$$PSNR = 10\log_{10}\left(\frac{f_{\max}^2}{MSE}\right), \tag{8.5}$$

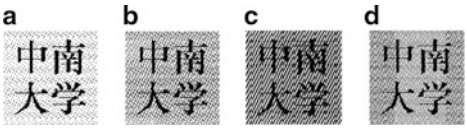

Fig. 8.2 Extracted watermarks from watermarked images cropped by (**a**) 10%, (**b**) 30%, (**c**) 40% and (**d**) 50%, respectively

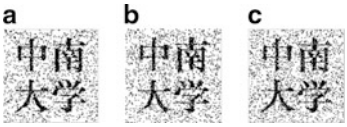

Fig. 8.3 Extracted watermarks from watermarked images with the lowest bits of 2, 3, 4 were damaged, respectively

where *MSE* denotes mean square error which is given by

$$MSE = \frac{1}{MN}\sum_{x=1}^{M}\sum_{y=1}^{N}\left(f(x,y) - f'(x,y)\right)^2, \tag{8.6}$$

where x and y are the image coordinates, M and N are the dimensions of the image, $f'(x,y)$ is the generated stego-image and $f'(x,y)$ is the host image. Also $f_{\max}$ is defined as the actual maximum value in the image. *PSNR* is often expressed on a logarithmic scale in decibels (dB). *PSNR* values falling below 30 dB indicate a fairly low quality, i.e., distortion caused by embedding can be obvious; however, a high quality stego-image should strive for 40 dB and above. In the proposed scheme, the *PSNR* between watermarked image and original image is 44.89 dB. Therefore, there is no obvious perceptual distortion between watermarked image and original one, the embedded watermark does not degrade the quality of original host image.

The normalized similarity (*NC*) is used to value the difference between original watermark and the extracted watermark, which is defined as:

$$NC = \frac{\sum_{x=1}^{M}\sum_{y=1}^{N} w(x,y) \times w'(x,y)}{\sum_{x=1}^{M}\sum_{y=1}^{N} w(x,y) \times w(x,y)} \tag{8.7}$$

Where $w(x,y)$ and $w^{'}(x,y)$ are the original watermark and the extracted watermark, M and N are the dimensions of the image. In the proposed scheme, the *NC* between original watermark and the extracted watermark with right secret keys is 1.

To test the robustness of the proposed scheme, some typical signal processing operations, such as cut attack, bits damaged attack and salt-pepper noise pollution are performed. Experiment results are shown in Figs. 8.2–8.4. It can be seen from Figs. 8.2–8.4 that the watermark can be extracted under various attacks.

Fig. 8.4 Extracted watermark from the watermarked image attacked by salt-pepper noise. (**a**) The watermarked image attacked by salt-pepper noise with 0.03 intensity; (**b**) the extracted watermark

8.1.1.4 Summary

In this subsection, a novel watermarking algorithm based on chaotic maps is proposed. One map is used to encrypt the embedding position and another one is used to determine the pixel bit for embedding in host image. The extraction of watermark depends only on private key, so it is a blind watermarking scheme. Simulations have confirmed that this new scheme is highly robust against various signal processing operations. Furthermore, it is also a secure scheme, only the one with the correct key can extract the watermark signal.

8.1.2 Chaos-Based Image DWT Domain Robust Digital Watermarking Algorithm

In this subsection, a novel meaning binary image digital watermarking algorithm based on chaos and DWT is proposed. The proposed scheme utilized chaotic sequences generated by general cat map to scramble the watermarks embedding positions, so that the security of embedding watermarks was enhanced. The improved binary operation methods were utilized, and watermarks are embedded to the low frequency coefficients of deep wavelet domain. The original image is not required for detecting the watermarks. The experimental results demonstrate that the embedded watermarks are invisible and this proposed watermarking algorithm is robust against many common attacks.

8.1.2.1 Wavelet Decomposition to Images

Discrete wavelet transform (DWT) is a novel kind of signal analysis theory rising in recent years. The basic idea of wavelet transform is do frequency separation carefully to signals, namely, multi-resolution analysis. Figure 8.5 shows the principle of a second-level wavelet decomposition. Through the first-level wavelet decomposition, the original image is decomposed into four first-level subimages: a low frequency subimage LL1, a horizontal detail subimage HL1, a vertical detail subimage LH1 and a diagonal detail subimage HH. LL1 is the proximate subimage of the original image that can be decomposed ulteriorly. If we perform wavelet

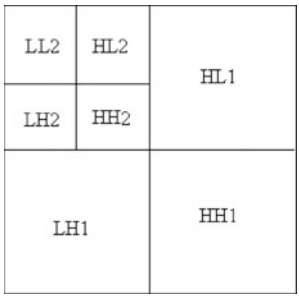

Fig. 8.5 A second-level wavelet decomposition

decomposition operation to the low frequency subimage LL1 again, then we can obtain four lower resolution second-level subimages: LL2, HL2, LH2 and HH2. To repeat the decomposition operation, one can do multi-level wavelet decomposition to the image. The lowest frequency subimage concentrates most of energy of the original image, which depicts the main features of the original image and is called the proximate subimage of the original image. The lowest frequency subimage possesses the strongest ability to resist outside interference. While every high frequency subimage, called the detail subimages of the original image, holds the edge and details of each direction, and they depict the edge and details features of the original image. However, high frequency subimages are vulnerable to outside interference and have poor stability.

8.1.2.2 Watermark Embedding and Extraction algorithm

(1) Watermark embedding

To denote a binary watermark image by $W = \{w(x,y),\ 1 \le x \le M_1,\ 1 \le y \le M_2\}$, where M_1 and M_2 is the width and height of the binary watermark image, respectively. The original host image is denoted by $F = \{f(x,y),1 \le x \le N_1,1 \le y \le N_2\}$, where N_1 and N_2 is the width and height of the host image, respectively. For simplicity we assume that $M_1 = M_2 = M$, $N_1 = N_2 = N$. We use the generalized cat map (8.3) to scramble the positions of watermarks, and embed watermarks in the wavelet coefficients of host image.

First, we define two terms: for a binary image, the pixel value in majority is called dominant watermark value, and the pixels that have dominant watermark value are called dominant pixels. While the pixel value in minority is called inferior watermark value, and the pixels that have inferior watermark value are called inferior pixels. Before watermark embedding, we scan the binary watermark image to find the dominant watermark value or the inferior watermark value. When embedding a dominant pixel watermark to a wavelet coefficient of the host image, we modify the wavelet coefficient value such that the fraction part of the wavelet coefficient value is changed into the range $[d_1, d_2]$. When embedding a inferior pixel watermark to a wavelet coefficient of the host image, we modify the wavelet coefficient value such that the fraction part of the wavelet coefficient value

is changed to zero. We call d_1 and d_2 lower threshold value and upper threshold value, respectively, and call $d = d_2 - d_1$ distance of threshold value.

Second, the watermark embedding algorithm based on DWT and chaotic scrambling is described as follows.

Step 1: Set the system parameters a, b, c in the generalized cat map and the iterate times n.

Step 2: Wavelet transform. Do a fourth-level wavelet decomposition to the host image.

Step 3: Scan the binary watermark image to find the dominant watermark value b_0, and scan the fourth-level subimage LL4 of the cove image to find the average value of the decimal part of wavelet coefficients c_0. Set the lower and upper threshold value of the fraction d_1 and d_2 such that c_0 is the median of $[d_1, d_2]$.

Step 4: For all watermark pixels $w(x, y)$, $x = 1 \sim M$, $y = 1 \sim M$, do following operations:

- Generate the embedding position (x', y') in the low frequence subimage of host image for watermark pixels $w(x, y)$ by using the generalized cat map, then determine the decimal part of wavelet coefficient $c(x', y')$: $c(x', y') = f(x', y') - floor(f(x', y'))$. Where $floor(x)$ rounds the elements of x to the nearest integers towards minus infinity.
- Embed watermark $w(x, y)$ into wavelet coefficient $f(x', y')$ using flowing rules:

if $w(x, y) == b_0$ & $c(x', y') < d_1$
  $f(x', y') = floor(f(x', y')) + d_1$;
else if $w(x, y) == b_0$ & $c(x', y') > d_2$
  $f(x', y') = floor(f(x', y')) + d_2$;
end if
if $w(x, y) == 1 - b_0$ & $c(x', y') < 0.5$
  $f(x', y') = floor(f(x', y'))$;
else if $w(x, y) == 1 - b_0$ & $c(x', y') >= 0.5$
  $f(x', y') = f(x', y') + 1 - c(x', y')$;
end if

Step 5: Do inverse wavelet transformation to obtain the watermarked image.

(2) Watermark extraction

The watermark extraction algorithm is described as follows.

Step 1: Wavelet transform. Do a fourth-level wavelet decomposition to the watermarked image to obtain detail subimages with different resolution and one lowest frequency subimage LL4.

Step 2: For all positions (x, y), $x = 1 \sim M$, $y = 1 \sim M$, do following operations:

- Determine the watermark embedding position (x', y') corresponding to watermark $w(x, y)$ via the generalized cat map, then Determine the decimal part of wavelet coefficient $c'(x', y')$ in subimage LL4: $c'(x', y') = f'(x', y') - floor(f'(x', y'))$.

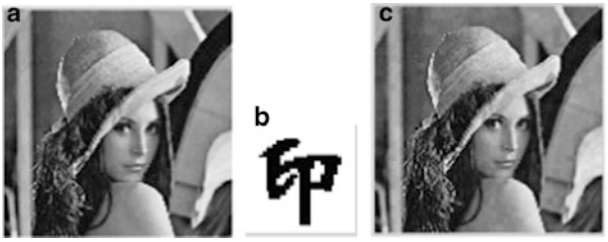

Fig. 8.6 The experimental images. (**a**) "Lena" host image; (**b**) the watermark image; (c) the watermarked image

Table 8.1 PSNR corresponding to different range [d1, d2]

[d1, d2]	PSNR
[0.25,0.75]	39.7135
[0.30,0.90]	39.7599
[0.10,0.70]	39.9274
[0.15,0.75]	40.3058
[0.20,0.80]	40.4120
[0.25,0.85]	40.2198

- According to the decimal part of wavelet coefficient $c(x', y')$, recover watermark value $w'(x, y)$ by using flowing rules:

if $c'(x', y') \leq d_1/2 | c'(x', y') \geq d_2 + (1-d_2)/2$
 $w'(x, y) = 1-b_0$;
else
 $w'(x, y) = b_0$;
end

8.1.2.3 Experimental Results

To demonstrate the effectiveness of the proposed algorithm, MATLAB simulations are performed by using the 512 × 512 pixels gray level Lena image and the 32 × 32 pixels binary watermark logo "印". In our algorithm, three independent parameters of Arnold cat map are set as $a = 1$, $b = 2$, $c = 3$, respectively. The iteration number $n = 7$. Set $[d_1, d_2] = [0.40, 0.60]$. Figure 8.6 shows the original host image, watermark image and the watermarked image, respectively. First, some values of *PSNR* corresponding to different range $[d_1, d_2]$ are calculated, which are shown in Table 8.1.

Then, set $[d_1, d_2] = [0.40, 0.60]$ (*PSNR* = 36.7880), To test the robustness of the proposed algorithm to some common attacks. Table 8.2 shows the *NC* values under different attacks.

Figures 8.7–8.11 show the watermarked images received various attacks and the related watermarks extracted from the watermarked images.

Table 8.2 NC values under different attacks

JPEG compression	Quality parameters =12 %	Quality parameters = 10 %	Quality parameters = 8 %
NC	1	0.9886	0.9557
Gaussion noise	Variance = 0.01	Variance = 0.02	Variance = 0.03
NC	0.9772	0.9215	0.8506
Salt and pepper noise	Density = 0.02	Density = 0.03	Density = 0.04
NC	0.9835	0.9430	0.9139
Median filtering	10 times 3 × 3	10 times 5 × 5	1 times 7 × 7
NC	0.9899	0.8987	0.9582
Image cut	Middle cut 14 %	Below cut 20 %	Top cut 49 %
NC	0.8532	0.8025	0.5139

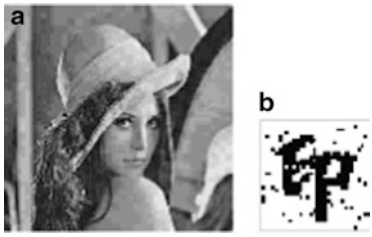

Fig. 8.7 JPEG compression. (**a**) The watermarked image with JPEG compression quality parameters dropped to 8%; (**b**) the related watermark extracted from the watermarked image (**a**)

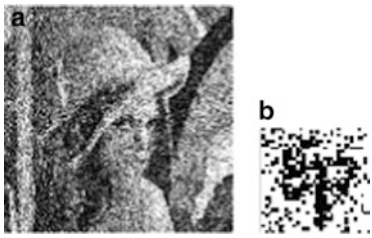

Fig. 8.8 Gaussion noise. (**a**) The watermarked image added Gaussion noise with average value 0 and variance 0.03; (**b**) the related watermark extracted from the watermarked image (**a**)

① JPEG compression

When the quality parameters are varied from 100% to 12%, the values of normalized similarity (*NC*) between original watermark and the extracted water mark are all equal to 1. When the quality parameters dropped to 8%, the watermarked image and the related watermark extracted from the watermarked image are shown in Fig. 8.7a, b, respectively.

② Gaussion noise

When the watermarked image is added Gaussion noise with average value 0 and variance 0.01, 0.02, and 0.03, respectively, all the watermarks can be extracted rightly. Figure 8.8a shows the watermarked image added Gaussion noise with

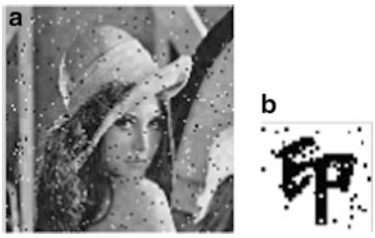

Fig. 8.9 Salt and pepper noise. (**a**) The watermarked image added salt and pepper noise with density 0.03; (**b**) the related watermark extracted from the watermarked image (**a**)

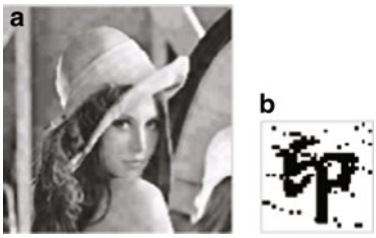

Fig. 8.10 Median filtering. (**a**) The watermarked image performed one times median filtering perations with window 7 × 7; (**b**) the related watermark extracted from the watermarked image (**a**)

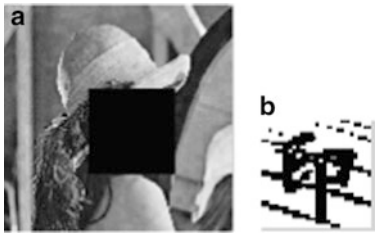

Fig. 8.11 Image cut. (**a**) The watermarked image is cut an area of 14.06 % on the central part; (**b**) the related watermark extracted from the watermarked image (**a**)

average value 0 and variance 0.03, while Fig. 8.8b shows the watermark extracted from the watermarked image.

③ Salt and pepper noise

When the watermarked image is added salt and pepper noise with density 0.02, 0.03, and 0.04, respectively, all the watermarks can be extracted rightly. Figure 8.9a shows the watermarked image added salt and pepper noise with density 0.03, while Fig. 8.9b shows the watermark extracted from the watermarked image.

④ Median filtering

When the watermarked image is performed one times median filtering operations with window 3 × 3, 5 × 5, and 7 × 7, respectively, or is performed ten times median filtering operations with window 3 × 3, all the watermarks can be extracted rightly. Figure 8.10a shows the watermarked image performed one times median filtering operations with window 7 × 7, while Fig. 8.10b shows the watermark extracted from the watermarked image.

⑤ Image cut
When the watermarked image is cut an area of 20 % on different sites, all the watermarks can be extracted rightly. Figure 8.11a shows the watermarked image that is cut an area of 14.06 % on the central part, while Fig. 8.11b shows the watermark extracted from the watermarked image.

8.2 Chaos-Based Fragile or Semi-Fragile Image Watermarking

With the development of digital technologies and Internet technologies, digital contents can be easily accessed via different transmission channels, such as Internet, wireless networks. Due to advantages of digital contents, perfectly reproduced and easily modified, many issues have become more urgent than in analog times, such as copyright protection and content authentication. Specific speaking, from a points of view of content providers, they want to present or sell their products through networks or other channels without threats of any pirate. Hence, content providers are eager to find some novel measures to protect theirs products. Similarly, after receiving a piece of digital content from legitimate channel, users expect that their received product is just from the real seller and is the original products without any modified. Hence, there should be an authentication mechanism that supports the security of this kind action and its manner.

Watermark can be classified into two types according its function, namely robust watermark for copyright protection and fragile watermark for integrity verification. In this section, we focus only on fragile watermarking, which can authenticate those multimedia received by unreliable channels, such as public Internet and wireless networks. Of course, fragile watermark also have many other functions. For example, it can be used to maintain the integrity of medical data in an electronic format. Nowadays, many fragile watermarking schemes for content authentication and integrity verification have been proposed. Generally speaking, all fragile watermark schemes can be classified into semi-fragile [7] and complete fragile schemes [8] according to integrity criteria. Authentication based on semi-fragile watermarking is also called soft authentication [9], which allows accepting certain content modifying without going beyond predefined quality or semantic meaning level, such as JPEG compression.

8.2.1 Chaos-Based Fragile Watermarking Algorithm for Image Authentication

In this subsection, a fragile watermarking scheme for image authentication is presented. Giving the initial value of the chaotic mapping, the fragile watermarks are obtained by iterating several times of the chaotic mapping. The watermarks are then inserted into the two least significant bit (LSB) planes of the host image. The proposed scheme can localize the tampers on the watermarked image by using the

high sensitivity on initial value of the chaotic mapping. Moreover, no host image is needed in watermarks extraction. The simulation results show the effectiveness, sensitivity and localization of tamper-proofing of this algorithm.

8.2.1.1 The Watermarking Algorithm

(1) Generating chaotic watermarks

In recent years, the security issues about watermarking algorithms have received increasing attention. Hence, chaotic maps are used to generate fragile watermarks for increasing the security. The reason is that the sequences generated by chaotic maps are very sensitive to initial conditions, in the sense that two chaotic sequences generated from different initial conditions are uncorrelated statistically. Logistic map is one of the simplest chaotic maps, which is determined by (8.4). It has been shown that this equation shows period doubling route to chaos in the system parameter μ range 1–4. While $\mu > 3.83$, the system is chaotic. In general, the value of μ is selected closer to 4. Here, we set that μ and z_0 equal to 4 and 0.238, respectively. A chaotic sequence using μ and z_0 is generated. By changing the chaotic sequence into matrix $\mathbf{R}$ with size equal to original digital image. And finally, the element of $\mathbf{R}$ is mapped to the integers 0, 1, 2 or 3 so that we generate a chaotic watermark image. We denote the chaotic watermark image by $\mathbf{W}_C$, therefore each element of $\mathbf{W}_C$ is an integer in the range 0–3.

(2) The watermark embedding algorithm

Before describing the proposed algorithm, we state some assumptions as follows. The size of an original image $\mathbf{A}$ is $M \times M$, and the size of watermark $\mathbf{W}_C$ has the same size with the host image $\mathbf{A}$. Set the two LSB planes of $\mathbf{A}$ as zero, get $\mathbf{A}_0$. Then for each pixel in image $\mathbf{A}_0$, add the corresponding element value of chaotic watermark matrix $\mathbf{W}_C$ to the pixel-value. Finally we obtain the watermarked image $\mathbf{A}_1$, i.e.

$$\mathbf{A}_1(i,j) = \mathbf{A}_0(i,j) + \mathbf{W}_c(i,j) \tag{8.8}$$

Where $i, j = 1, 2, 3, \ldots, M$. The detail steps of the algorithm are described as follows:

Step 1: Input $M \times M$ image matrix $\mathbf{A}$.
Step 2: Input the initial condition z_0 and system parameter μ of Logistic map, and secret Integer N.
Step 3: The Logistic map is iterated N times.
Step 4: Then Logistic map is iterated $M \times M$ times, delete the first N values of the chaotic sequence, then we obtained a chaotic sequence that the length is $M \times M$. Store the chaotic sequence to a one dimensional array $\mathbf{R}_0$.
Step 5: Reshape the chaotic sequence $\mathbf{R}_0$ into matrix $\mathbf{R}$ with size of $M \times M$.

Step 6: Transform the matrix $\mathbf{R}$ into integer matrix $\mathbf{W}_C$ such that each element value of $\mathbf{W}_C$ is a integer in the range [0,3], the method of transformation is as follows:

If $\mathbf{R}(i, j) < 0.25$ then
    $\mathbf{W}_C(i, j) = 0$;
ElseIf $\mathbf{R}(i, j) < 0.5$ then
    $\mathbf{W}_C(i, j) = 1$;
ElseIf $\mathbf{R}(i, j) < 0.75$ then
    $\mathbf{W}_C(i, j) = 2$;
Else
  $\mathbf{W}_C(i, j) = 3$;
End if

Step 7: Set the two LSB planes of $\mathbf{A}$ as zero, get $\mathbf{A}_0$.
Step 8: For one pixel of $\mathbf{A}_0$ in the position (i, j), we denote the pixel-value by $\mathbf{A}_0(i, j)$. Do the following calculation: $\mathbf{A}_1(i, j) = \mathbf{A}_0(i, j) + \mathbf{W}_C(i, j)$.
Step 9: For another pixel of $\mathbf{A}_0$, repeat the operation of Step 8, until every pixel in $\mathbf{A}_0$ is processed. Then, the watermarked image $\mathbf{A}_1$ is obtained.

(3) The extracting and authentication algorithm

The watermark extracting process is almost the same as the watermark embedding but with reverse operations. Firstly, similar to the watermark embedding procedure, the same chaotic integer matrix $\mathbf{W}_C'$ is generated by z_0 and μ of Logistic map. Secondly, get the watermark $\mathbf{W}_1$ form the watermarked image $\boldsymbol{A}_1$ by extracting the two LSB planes.

Having obtained the two versions of fragile watermark, $\mathbf{W}_1$ and $\mathbf{W}_C'$, we define the tamper detection matrix:

$$T = |W_1 - W_C\prime| \tag{8.9}$$

Then, from $\mathbf{T}$, we can locate the modified area of watermarked image. If $\mathbf{W}_1 = \mathbf{W}_C'$ then $\mathbf{T} = 0$. This means that the watermarked image was not tampered. Otherwise, the "1" elements in the tamper detection matrix denote the pixels that were tampered. Our proposed scheme is blind because the original image is not required during the watermark detection process.

8.2.1.2 Experimental Results

Generally speaking, watermark algorithm can be evaluated from many factors. But there exists some differences between robust watermark and fragile watermark. For robust watermark, the most important factor is the robustness of algorithm resisting all kinds of attacks, such as common signal processing operators, geometric attacks and so on. The second important factor is perceptual quality. Watermarked image should not be perceptual by human visual system. However,

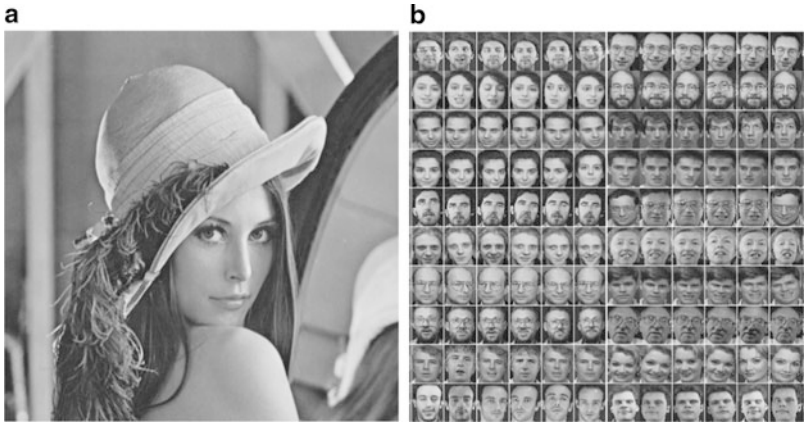

Fig. 8.12 The host image. (**a**) Lena host image; (**b**) Face host image

for fragile watermark, the robustness is not the most important factor again. The most important factors include:

(1) Perceptual quality, which is similar to robust watermark. All kinds of processes do not degrade the quality of original image. Otherwise, this process is not suitable, and it will not be accepted by users.
(2) Location capability, which is the capability that algorithm can locate the modified region in image. There are three kinds of locating accuracy, namely locating single-pixel tampered, locating single-block tampered and locating the whole signal.
(3) Security, which is a very important factor. The goal of fragile watermark is to guarantee the authenticity and integrity of received image. If unauthorized user can forge arbitrary image, the fragile watermark system is nonsense. Hence, security issue should be considered carefully by designer in constructing the fragile watermark system.

In this scheme, we evaluate the algorithm by extensive experiments. The host image used in this scheme is a pixel of gray-level with 256 intensity distribution, and of a size containing 512 × 512 pixels, as shown in Fig. 8.12a, b. The chaotic watermark image, which is of four level image with a pixel of 4 intensity distributions, is shown in Fig. 8.13. It is generated by the Logistic map generator with $z_0 = 0.238$, $\mu = 4$, $N = 1{,}000$, and each of its element is scaled into the rangee [0, 3]. Figure 8.14a, b are the watermarked images by embedding Fig. 8.13 into the two LSB bit planes of Fig. 8.12a, b.

From Fig. 8.14, we find that the quality of watermarked image is very good, and the posted numbers are measured by the criterion estimation of peak-signal-to-noise ratio (*PSNR*). In fact, from the theory analysis, fragile watermark only use the two least significant bit planes to embed information, hence, the worst *PSNR* is

$$PSNR_{worst} = 10 \times \log_{10}(255^2/3^2) = 38.5884 \tag{8.10}$$

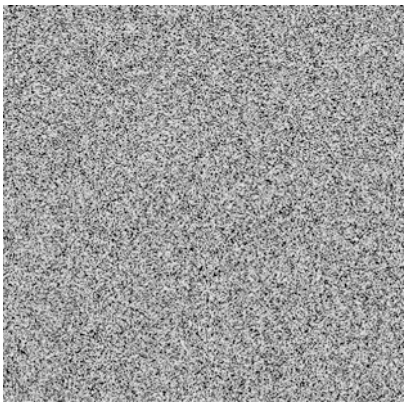

Fig. 8.13 The chaotic watermark image

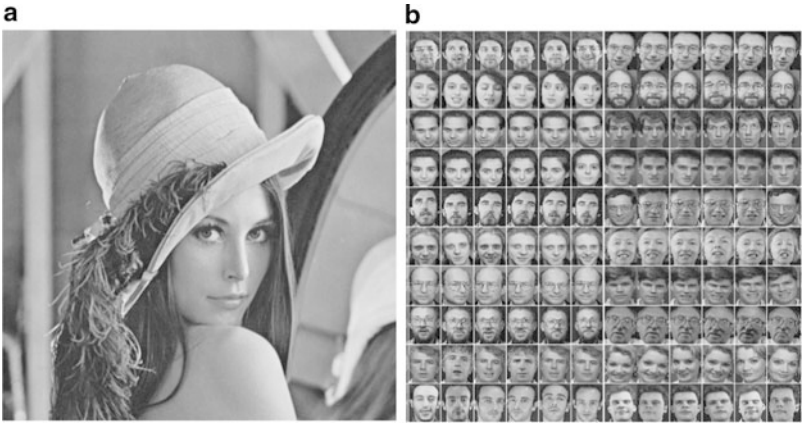

Fig. 8.14 The watermarked image. (**a**) The watermarked image of Lena; (**b**) The watermarked image of Face

And the 3^2 in this formula indicates that each pixel value will be changed as following: $0 \rightarrow 3$, and $3 \rightarrow 0$. However, in our scheme of chaotic watermark generating algorithm, there are four cases of pixel value change: 0, 1, 2 and 3. According to the random distribution assumption, the distributions of 0, 1, 2 and 3 in the process are nearly equal-probability, Hence, the *PSNR* on average is the case where the probability of each case of pixel value change is 1/4. So the average *PSNR* is:

$$PSNR_{aver} = 10 \times \log_{10}\{255^2/[(0^2 + 1^2 + 2^2 + 3^2) \times 0.25]\} = 42.6901 \quad (8.11)$$

Fig. 8.15 Two tampered watermarked images. (**a**) The tampered Lena watermarked image; (**b**) the tampered Face watermarked image

From Fig. 8.14a, we obtain $PSNR_{\text{aver}} = 43.2640$ dB. From Fig. 8.14b, we obtain $PSNR_{\text{aver}} = 43.5497$ dB, which are very close to theory value 42.6901. And this indicates that the resulting watermark image generated by a Logistic map generator is randomly distributed.

To show the capability of locating the tamped area of using the proposed scheme, we first modified the watermarked image. Figures 8.15 and 8.16 show the capability of locating the tampered area in watermarked image.

Figure 8.15a, b are the tampered image of Fig. 8.14a, b created by using the software Photoshop and Microsoft paint. In Fig. 8.15a, the tampered areas include two parts: the first part corresponds to the face of Lena image in Fig. 8.14a replaced by the face of Elaine image, and the second part is the text "WHO IS ?" and "Elaine?" in Fig. 8.15a. It should be noted that the text can be classified into two classes. The first class is "WHO IS ?" text, it is pure white color, and the second class text is "Elaine?" text, it is pure black color. In Fig. 8.15b, the modified area is the second row at the left and the fifth row at the right, we change the position of them. However, due to the similarity, we can not find the remarkable difference between the Figs. 8.14b and 8.15b. In order to locate the modified area, according to extracted watermarks from Fig. 8.15a, b, we calculate the tamper detection matrix **T** to get the locations of modified areas. As observed from Fig. 8.16a, b, we can locate accurately the tampered areas.

8.2.1.3 Summary

In this subsection, we proposed a fragile watermark scheme based on chaotic map and substituting the low least significant bit planes in spatial domain of host image, which is efficient, secure and blind (the source image is not needed). In this scheme, the chaotic map are used to generate the fragile watermark in order to improve the

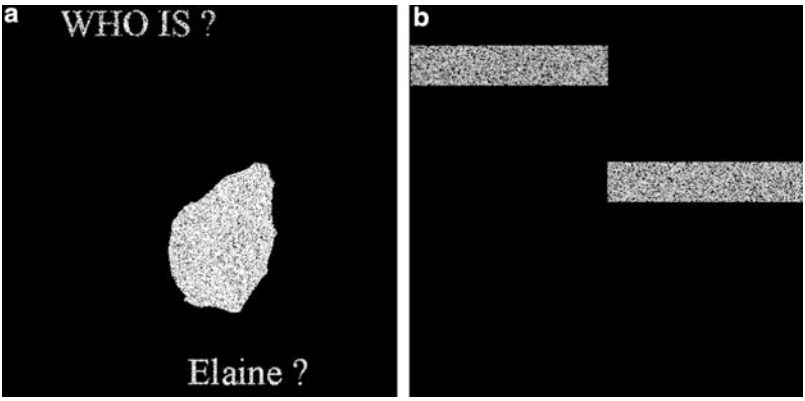

Fig. 8.16 Localization of tampered area in watermarked images. (**a**) Localization of tampered area in Figs. 8.8–8.14a; (**b**) localization of tampered area in Figs. 8.8–8.14b

security of the scheme. Furthermore, the presented method can also detect the exact tampered areas on the watermarked image. Experiments results have confirmed that this scheme has high fidelity and excellent capability of localizing modified areas in watermarked image. Therefore, it is a promising technique for fragile watermarking for high-quality and reliable still images.

8.2.2 Chaos-Based Semi-Fragile Image Watermarking

This subsection presents a DCT-based semi-fragile watermarking scheme, which exploiting the features of block DCT and chaotic map. The original image is divided into many 4×4 blocks and perform block DCT. For each block, the value of DCT DC coefficient A(0,0) is mapped as the initial value of chaotic map. After iterating several times, the binary watermark is obtained and is embedded into the DCT AC coefficient A(1,0). Its application includes tamper detection, image verification and copyright protection of multimedia content. Experiment results show the scheme can detect and localize malicious attacks with high peak signal-to-noise ratio (*PSNR*), while tolerating certain degree of JPEG compression and channel additive white Gaussian noise (AWGN).

Digital watermark can be classified as robust watermark, fragile watermark and semi-fragile watermark. The robust watermark survives when the watermarked digital content is severely attacked and thus can be applied in copyright protection. On the other hand, the fragile watermark will be destroyed even if the change in the marked digital media is minute. By reason of this property, image authentication becomes a prospective application of it. There are two different kinds of operations for images, one kind is mild modification such as JPEG compression and channel additive white Gaussian noise (AWGN), another is malicious attack such as feature

replacement or content adding. As a tradeoff of robustness and fragility, semi-fragile watermark that can resist "content preserving" operations and be sensitive to "content altering" transforms is more practicable than fragile watermark in image authentication. In [10], the authors alter the wavelet domain coefficients according to the binary watermark stream to meet the statistical parameter equity relationship based on the Gaussian mixture statistical model. They tell mild modification from malicious attack through parameter trends. But they treat AWGN as malicious attack. In [11], the authors present a fragile watermarking combine the time-frequency feature of DWT with the chaotic map. Here, we propose a DCT-based chaotic semi-fragile watermarking scheme aiming at better performance.

8.2.2.1 The Proposed Scheme

(1) Chaotic map

Chaos is a special solution for non-linear equation. Its random output is decided by a determinate equation. We can control a set of parameters of chaotic system as the key of the proposed scheme. Many research on this issue shows that it is more secure than traditional ways to generate a pseudo-random sequence. Consider a 1D discrete chaotic map f: $U \rightarrow U$, $U \subset R$:

$$z(n+1) = f(\lambda, z(n)),\ \lambda \in R, z(n) \in U \tag{8.12}$$

Where $n = 0,1, \ldots$ denotes the number of iteration, λ controls the dynamic behavior of the chaotic map. A quantization function can make $z(n) \in \{0,1\}$ after iteration. Given slightly two different initial values of chaotic map will lead to two totally different pseudo-random sequences after many times iteration. Security is obtained by such high sensitivity characteristics of chaotic map. Here, we applied Logistic chaotic map to generate binary pseudo-random sequences as semi-fragile watermarks.

(2) Watermark embedding

Unlike usual watermarking scheme, watermark is created according to the host image in our algorithm. The binary watermark is embedded into the host image by using a simple odd-even quantization. Define odd-even quantization function:

$$x' = g(x, w, Q),\ x \in R,\ w \in \{0, 1\},\ Q \in Z^{+} \tag{8.13}$$

Which performs quantization on x into odd-even region according to a binary number w, Q is the quantization parameter, and x' is obtained as follows:

$$x' = \begin{cases} [|x|/\mathrm{Q}] \times Q \times \mathrm{sgn}(x), & if\ w = 0 \\ ([|x|/\mathrm{Q}] + 0.5) \times Q \times \mathrm{sgn}(x), & if\ w = 1 \end{cases} \tag{8.14}$$

Where $\mathrm{sgn}(x) = 1$, if $x \geq 0$; $\mathrm{sgn}(x) = -1$, if $x \leq 0$. $[\cdot]$ denotes the round function. Watermark embedding algorithm is described as follows:

Step 1. Select parameters: K, Q_1, Q_2, and N. K is the private key of the scheme. Q_1 and Q_2 are the quantization parameters. $N \times N$ is the size of image block.
Step 2. Divide the original image into $L_1 \times L_2$ blocks $\mathbf{B}_{ij}$, $i = 1, 2, \ldots, L_1$; $j = 1, 2, \ldots, L_2$, $L_1 = M_1/N$, $L_2 = M_2/N$, where $M_1 \times M_2$ is the size of original image.
Step 3. Create two-dimensional pseudo-random arrays n_{ij} ($1 \leq i \leq L_1$, $1 \leq j \leq L_2$) using key K.
Step 4. Perform following operations on each block of the image:

For $i = 1$ to L_1
For $j = 1$ to L_2

Perform DCT operation on $\mathbf{B}_{ij}$: $\mathbf{A} = \mathrm{DCT2}(\mathbf{B}_{ij})$. Map$Q_n = \lfloor \mathrm{A}(0,0)/Q_1 \rfloor$ as the initial value of the chaotic map; after iterate n_{ij} times, we obtain $w(i, j) \in \{0, 1\}$. Embed $w(i, j)$ into $\mathbf{A}(1,0)$ to get $\mathbf{A}'(1,0)$: $\mathrm{A}'(1,0) = \mathrm{g}(\mathrm{A}(1,0), w(i,j), Q_2)$. Perform inverse DCT operation on $\mathbf{A}'$ to get $\mathbf{B}'$: $\mathbf{B}'_{ij} = \mathrm{IDCT}(\mathbf{A}')$.

(3) Watermark detection

The Watermark detection algorithm is described as follows:

Step1-Step3. similar to the Steps 1–3 of the watermark embedding process.
Step 4. Perform the following operations on each block of the possible watermarked image:

FOR $i = 1$ to L_1
FOR $j = 1$ to L_2

Perform DCT operation on $\mathbf{B}'_{ij}$: $\mathbf{A}' = \mathrm{DCT2}(\mathbf{B}'_{ij})$.
Map $Q_n = \lfloor \mathbf{A}(0,0)/Q_1 \rfloor$ as the initial value of the chaotic map; after iterate n_{ij} times, we obtain $w_1(i, j) \in \{0, 1\}$. On the other hand, we may obtain another version of extracted watermark from the quantized AC coefficient, say $\boldsymbol{A}'(1,0)$, it is as follow:

$$w_2(i,j) = \begin{cases} 1, \ \textit{if } Q/4 \leq |\mathrm{A}(1,0)| \ \textit{Mod } Q < 3Q/4 \\ 0, \ \textit{else} \end{cases} \tag{8.15}$$

Step 5. Having obtained the two versions of extracted watermark, calculate the tamper detection matrix: $\mathbf{T} = |\mathbf{W}_1 - \mathbf{W}_2|$. Where $\mathbf{W}_1 = \{w_1(i,j), 1 \leq i \leq L_1, 1 \leq j \leq L_2\}$, $\mathbf{W}_2 = \{w_2(i,j), 1 \leq i \leq L_1, 1 \leq j \leq L_2\}$.

If $\mathbf{W}_1 = \mathbf{W}_2$ then $\mathbf{T} = 0$. This means that the watermarked image was not tampered. Otherwise, the "1" elements in the tamper detection matrix denote the

pixels that were tampered. Note that the size of tamper detection matrix is $(M_1/N) \times (M_2/N)$, which is about $1/(N \times N)$ of the host image. Thus one element in the matrix denotes a corresponding $N \times N$ block in the host image. The proposed scheme is blind because the original image is not required during the watermark detection process.

Since the scheme is designed to be robust to mild modification in all cases, it is inevitable that we do not detect all malicious attack in pixel-wise. For a practical case such as image-content-add attack, we assume that malicious attack always be applied in a certain region in host image. Thus tamper pixels are always continuous. For a certain tamper detection matrix element $\boldsymbol{T}(i, j)$, if the number of tampered element in the neighboring elements of $\boldsymbol{T}(i, j)$ is greater than a given threshold, we thus consider $\boldsymbol{T}(i, j)$ to be tampered, too. The following equation is the summary of such post-processing (PP) operation of tamper detection matrix:

$$T(i,j) = \begin{cases} 1, & \sum_{u=-\alpha}^{u=\alpha} \sum_{v=-\alpha}^{v=\alpha} T(i+u, j+v) > \beta \\ 0, & \sum_{u=-\alpha}^{u=\alpha} \sum_{v=-\alpha}^{v=\alpha} T(i+u, j+v) \leq \beta \end{cases} \tag{8.16}$$

8.2.2.2 Experiment Results and Analysis

To check the validity of the scheme, many attacks have been conducted. A $512 \times 512 \times 8$ "Peppers" gray-scale image is adopted as illustration the purpose. We map the integer gray-scale value range in [0,255] to the double class range [0,1]. We first perform block DCT of the 'Peppers' image with $N = 4$. Let $Q_1 = 0.282$, $Q_2 = 0.107$. Figure 8.17a shows the watermarked image with $PSNR = 42.61$ dB. We detract a pepper, add a pepper and a word "DNA" in the watermarked image which is shown in Fig. 8.17b. The tampered region is shown in Fig. 8.17c. It is show that three malicious tamper positions are all detected with relative precise localization.

If we consider the situation when the image is mildly modified such as severe AWGN while on the other hand it is maliciously attacked such as image content tamper, Fig. 8.18 shows the results of PP operation when using different parameters. It is very like the erosion operation in image processing. When $\alpha = 1, \beta = 3$, we can tell the malicious attack from the mild modification very clearly. The performance is evaluated by the normalized cross-correlation (NC) without PP operation and bit error rate (BER) of tamper detection matrix after PP operation.

Table 8.3 shows the NC value and BER after JPEG compression with different quality factor (QF) and AWGN with different variance σ^2.

In the proposed scheme, Q_1 partly controls the sensitivity and security of the proposed scheme. Q_2, controls the robustness together with Q_1. Larger Q_2, leads to more robustness. While smaller Q_2, leads to more fragility. Thus a tradeoff exists.

Fig. 8.17 Tamper detection. (**a**) The watermarked image. (**b**) Image after attack. (**c**) The tampered region

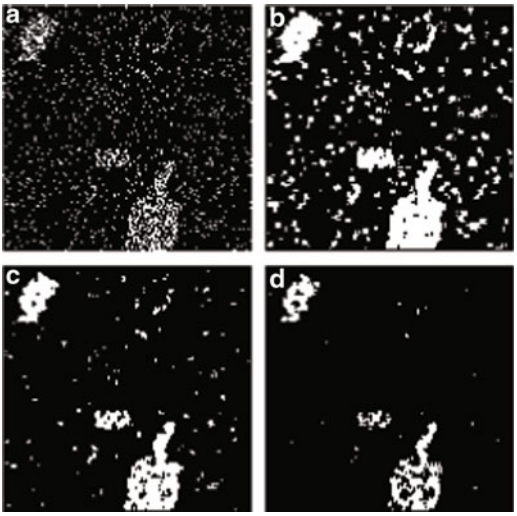

Fig. 8.18 PP operation on matrix T. (**a**) No PP operation; (**b**) $\alpha = 1$, $\beta = 1$; (**c**) $\alpha = 1$, $\beta = 2$; (**d**) $\alpha = 1$, $\beta = 3$

Table 8.3 Robust against JPEG compression and AWGN

JPEG:QF	NC	BER	AWGN:σ2	NC	BER
100	0.997	0.00	6	0.983	0.001
90	0.993	0.00	12	0.973	0.004
80	0.988	0.00	18	0.970	0.191
70	0.981	0.00	24	0.964	0.057
60	0.974	2.5×10^{-3}	30	0.960	0.132

Note that only Q_2 influences the *PSNR* of the watermarked image. The computational complexity is mainly decided by the total number of embedding position and the chaotic iteration number of each embedding position, which is $O(E(n_{ij}) \times L_1 \times L_2)$, where $E(\cdot)$ denotes the expectation operator.

8.2.2.3 Summary

A DCT-based chaotic semi-fragile watermarking is presented. The main characteristic of this scheme is as follows: (1) Robust to mild modification and fragile to malicious attack. (2) Relative precise localization. (3) High invisibility. (4) Low computational complexity.

8.3 Chaos-Based Multipurpose Image Watermarking

8.3.1 Introduction

Digital multimedia is largely distributed today over the Internet and via CD-ROM. However, the possibility of lossless and unlimited copies of digital contents is a major obstacle from the owner's viewpoint for entering the digital world. Copy protection and content authentication have, therefore, been the two most important issues in the digital world. Conventional cryptographic systems permit only valid key holders access to encrypted data, but once such data is decrypted, there is no way to track its reproduction or retransmission. Over the last decade, digital watermarking has been presented to complement cryptographic processes. In general, there are two types of invisible digital watermarks addressed in the existing literature: robust and fragile (or semi-fragile) watermarks. Robust watermarks are generally used for copyright protection and ownership verification because they are robust to nearly all kinds of image processing operations. In comparison, fragile or semi-fragile watermarks are mainly applied to content authentication and integrity attestation because they are fragile to most modifications.

But conventional watermarking algorithms are mostly designed for only one purpose, namely, for either copyright protection or content authentication. To fulfill multipurpose applications, multipurpose watermarking algorithms have been presented. Lately, several multipurpose watermarking schemes have been presented, such as the scheme based on vector quantization (VQ) structure [12]. But The computational complexity in [12] is high, since the vector quantization in their scheme needs code book design, training and a VQ encoder, and the peak signal-to-noise ratio (*PSNR*) in [12] is low.

In this section, we present a multipurpose digital image watermarking scheme based on discrete wavelet transform (DWT) and chaotic map, which can be applied to image authentication and copyright protection. In the proposed scheme, by performing 2-level DWT of the host image, the robust watermarks are embedded in the low frequency sub-band LL2 of the host image, and the semi-fragile watermarks, generated by chaotic map, are embedded in the high frequency sub-bands HL2, LH2 or HH2, and the embedding coefficients are selected randomly by secret key. A improved quantization method (IQM) is utilized to embed robust watermarks. Watermarks can be extracted without the original image. Simulation results demonstrate the effectiveness of our algorithm in terms of robustness and fragility.

8.3.2 The Proposed Watermarking Scheme

8.3.2.1 Chaotic Map

Chaotic maps can be used for digital watermarking to increase the security. The most attractive features of chaos in information hiding are its extreme sensitivity to initial conditions and the outspreading of orbits over the entire space. These special characteristics make chaotic maps excellent candidates for watermarking and encryption.

The skew tent chaotic map is one of the simplest chaotic maps, described by

$$z_{n+1} = \begin{cases} z_n/p, \ if \ z_n \in [0,p] \\ (1-z_n)/(1-p), \ if \ z_n \in (p,1] \end{cases} \tag{8.17}$$

Where $z_n \in [0, 1]$ is the state of the system ($n = 0, 1, 2,\ldots$), and $p \in (0, 1)$ is the control parameter. For any $p \in (0, 1)$, the piecewise linear map (8.17) has a positive Lyapunov exponent and thus is always chaotic. All the sequences generated by the skew tent map are very sensitive to initial conditions, in the sense that two sequences generated from different initial conditions are uncorrelated statistically. Moreover, all the orbits of the skew tent map are dense in the range of the map [0,1]. A quantization function $q(\cdot)$ can make $z_n \in \{0, 1\}$ after iteration. So, a binary sequence is generated and can be used as watermarks. Given slightly two different initial values of the chaotic map will lead to two totally different pseudo-random sequences after many times iteration. Security is obtained by such high sensitivity characteristics of chaotic map.

8.3.2.2 Watermark Embedding

Before describing the proposed algorithm, we make some assumptions. Let $\mathbf{A}$ be the host image with size $M_1 \times M_2$, let $\mathbf{W}_R$ and $\mathbf{W}_F$ be the binary robust and fragile watermark respectively. Here, a visually meaningful binary image with size $L_1 \times L_2$, is used as the robust watermark $\mathbf{W}_R$, and a chaotic sequence with length $(M_1/4) \times (M_2/4)$, is used as the fragile watermark $\mathbf{W}_F$. By performing 2-level DWT of the host image, we obtain one low frequency sub-band LL2 and three high frequency sub-bands HL2, LH2, HH2, respectively. Then the robust watermarks are embedded in the low frequency sub-band LL2 of the DWT domain, while the semi-fragile watermarks, generated by chaotic map, are embedded in the high frequency sub-bands HL2, LH2 or HH2 of the DWT domain.

To achieve optimized embedding effect, the binary watermark is embedded into the host image by using the improvement quantization method (IQM), which is shown in Fig. 8.19. Different from the usual quantization method (UQM), in our IQM scheme, two quantization parameters Q and q are introduced, and the original binary robust watermark sequence $\mathbf{W}_R$ is firstly scanned to find the majority bit value b ($b = 0$ or $b = 1$).To embed a watermark bit in the block of host image DWT

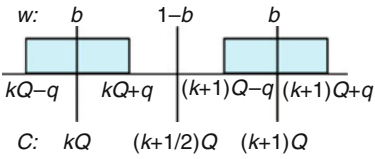

Fig. 8.19 The improvement quantization method

coefficient *f,* according to the watermark bit *w*, change the DWT coefficient *f* into different range. If the embedding watermark *w* is the majority bit value *b*, then force *f* to the nearest range $[kQ - q, kQ + q]$, if the embedding watermark *w* is 1–*b*, then force *f* to the nearest value $(k + 1/2)Q$, here, $k = 0, \pm1, \pm2, \ldots\ldots$.

The embedding algorithm for copyright protection watermark $\mathbf{W}_R$ is described as follows:

Step 1. Input parameters k_1,Q_1, q_1, and scan binary watermark image $\mathbf{W}_R$ to find the majority bit value *b*. Where $\mathbf{W}_R = \{w_r(i, j), i = 1,2, \ldots, L_{1;}\, j = 1,2, \ldots, L_2\}$.
Step 2. Perform 2-level DWT of the host image, then obtain four different frequency sub-bands, namely, LL2, HL2, LH2 and HH2.
Step 3. Permute the robust watermark image $\mathbf{W}_R$ with k_1 to obtain $\mathbf{W}_{PR}$. Where $\mathbf{W}_{PR} = \{w_{pr}(i, j), i = 1,2, \ldots, L_{1;}\, j = 1,2, \ldots, L_2\}$.
Step 4. Take one pixel bit $w_{pr}(i, j)$ from $\mathbf{W}_{PR}$, insert $w_{pr}(i, j)$ into the DWT coefficient of low $LL_2(i, j)$ frequency sub-bands with IQM.
Step 5. Repeat Step 4 until every pixel bit $w_{pr}(i, j)$ in $\mathbf{W}_{PR}$ is embedded.

Embedding algorithm for image authentication watermark $\mathbf{W}_F$ is described as follows:

Step 1. Input parameters k_2, Q_2, the initial value z_0 of chaotic map and system parameter *p*; and iterate the chaotic map 1,000 times in advance.
Step 2. iterate the chaotic map one times to obtain z_n, then transform z_n into 0 or 1 with quantization function $q(\cdot)$ to obtain fragile watermark bit $w_f(i, j)$: $w_f(i, j) = q(z_n)$.Where $q(x) = 0,1$.
Step 3. Select one coefficient randomly from $\{LH_2(i, j), HL_2(i, j), HH_2(i, j)\}$ by secret key k_2.
Step 4. Insert $w_f(i, j)$ into the selected coefficient in Step 3 with odd–even quantizition method.
Step 5. Repeat Step 2 to Step 4 until $(M_1/4) \times (M_2/4)$ watermark bits are embedded.

8.3.2.3 Watermark Detection and Image Authentication

For a possible watermarked image, different from the embedding processes, extract processes are the inverse operation of embedding processes.

The extracting algorithm for copyright protection watermark $\mathbf{W}_R'$ is described as follows:

Step 1. Input parameters k_1, Q_1, q_1, and the majority bit value *b*.

Step 2. Perform 2-level DWT of the watermarked image **A**', then obtain four different frequency sub-bands, namely, LL2', HL2', LH2' and HH2'.

Step 3. Extract one pixel bit $w'_{pr}(i, j)$ from the DWT coefficient $LL_2'(i, j)$ of low frequency sub-bands with inverse IQM, namely:

$$r = \left| LL_2'(i,j) \right| \ mod\ Q_1 \tag{8.18}$$

$$w'_{pr}(i,j) = \{b, if\ |r - Q_1/2| > (Q_1/4 - q_1/2)1 - b,\ else \tag{8.19}$$

Step 4. Repeat Step 3 until every watermark bit $w'_{pr}(i, j)$ is extracted. Then we obtain the permutated robust watermark matrix $\mathbf{W}_{PR}'$.

Step 5. Perform inverse permutation operation on $\mathbf{W}_{PR}'$ with k_1 to obtain the robust watermark image $\mathbf{W}_R'$.

Extracting algorithm for image authentication watermark $\mathbf{W}_F'$ is described as follow:

Step 1. Input parameters k_2', Q_2', and chaotic initial value z_0' and system parameter p' and iterate the chaotic map 1,000 times in advance.

Step 2. Iterate the chaotic map one times to obtain z_n', then transform z_n' into 0 or 1 with quantization function $q(\cdot)$ to obtain fragile watermark bit $w'_{f1}(i,j)$: $w'_{f1}(i,j) = q(z_n')$..

Step 3. Select a coefficient randomly from $\{LH_2'(i, j), HL_2'(i, j), HH_2'(i, j)\}$ by secret key k_2'.

Step 4. Extract one bit $w'_{f2}(i, j)$ from the selected coefficient in Step 3 with inverse odd–even quantizition method.

Step 5. Repeat Step 2 to Step 4 until $(M_1/4) \times (M_2/4)$ watermark bits are extracted, then the fragile watermark image $\mathbf{W}_{F1}$ and $\mathbf{W}_{F2}$ are obtained.

Step 6. Having obtained the two versions of fragile watermark, $\mathbf{W}_{F1}$ and $\mathbf{W}_{F2}$, we define the tamper detection matrix:

$$\mathbf{T} = |\mathbf{W}_{F1} - \mathbf{W}_{F2}| \tag{8.20}$$

If $\mathbf{W}_{F1} = \mathbf{W}_{F2}$ then $\mathbf{T} = 0$. This means that the watermarked image was not tampered. Otherwise, the "1" elements in the tamper detection matrix denote the pixels that were tampered. Note that the size of tamper detection matrix is $(M_1/4) \times (M_2/4)$, Thus one element in the matrix denotes a corresponding 4×4 block in the host image. The proposed scheme is also blind because the original image is not required during the watermark detection process.

Since our semi-fragile watermark scheme is designed to be robust to mild modification in all cases, it is inevitable that we do not detect all malicious attack in pixel-wise. For a practical case such as image-content-add attack, we assume that malicious attack always be applied in a certain region in host image. Thus tamper pixels are always continuous. For a certain tamper detection matrix element $\mathbf{T}(i, j)$, if the number of tampered element in the neighboring elements of $\mathbf{T}(i, j)$ is greater than a given threshold, we thus consider $\mathbf{T}(i, j)$ to be tampered, too. The following equation is the summary of such post processing (PP) operation of tamper detection matrix

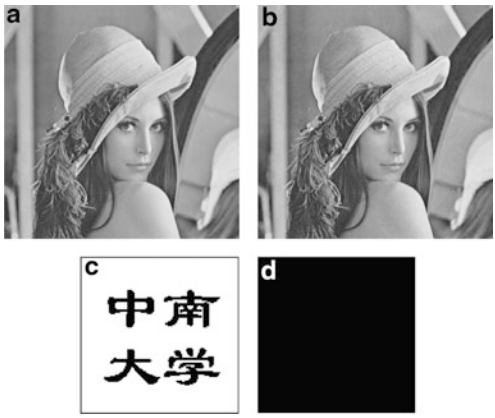

Fig. 8.20 Host image and watermark detection results. (**a**) Original host image; (**b**) watermarked host image; (**c**) extracted robust watermark; (**d**) tamper detection matrix

$$\mathbf{T}'(i,j) = \begin{cases} 1, & \sum_{u=-\alpha}^{\alpha} \sum_{v=-\alpha}^{\alpha} \mathbf{T}(i+u, j+v) > \beta \\ 0, & \sum_{u=-\alpha}^{\alpha} \sum_{v=-\alpha}^{\alpha} \mathbf{T}(i+u, j+v) \le \beta \end{cases} \tag{8.21}$$

8.3.3 Experimental Results

To evaluate the performance of the proposed method, the 512 × 512 Lena image with 8bits per pixel resolution is used for multipurpose watermarking. The Lena image is performed with 2-level discrete wavelet transform ($L = 2$). A binary watermark with size 128 × 128 is used for robust watermarking image $\mathbf{W}_{\mathrm{R}}$, namely, $L_1 = L_2 = 128$. Another 128 × 128 binary matrix, which obtained by the skew tent chaotic map, is used as semi-fragile watermark $\mathbf{W}_{\mathrm{F}}$. The size of host image, watermark **W**R and **W**F are equal to those in [12]. The pixel values of image is mapped in the range [0, 1]. Let $Q_1 = 0.2$, $q_1 = 0.05$; $Q_2 = 0.1$. Figure 8.20a shows the original host image, Fig. 8.20b shows the watermarked image with *PSNR* = 39.57 dB. It is noted that the *PSNR* is only 30.553 dB with the scheme in [12]. The extracted robust watermark image $\mathbf{W}_{\mathrm{R}}'$ and the tamper detection matrix **T** under non-attack is shown in Fig. 8.20c, d, respectively. In the robust watermark image, the majority bit value is $b = 1$ (white pixel). If the optimized embedding is not under considering, i.e., swap embedding positions for bit value b and $1 - b$ in Fig. 8.20, well then the *PSNR* is 35.68 dB.

We employ the normalized Hamming Similarity, *NHS*, to evaluate the effectiveness of the proposed algorithm. The *NHS* between the embedded binary watermark **W** and the extracted one $\mathbf{W}'$ is defined as

$$NHS = 1 - \frac{\mathrm{HD}(\mathbf{W}, \mathbf{W}')}{L_1 \times L_2}, \tag{8.22}$$

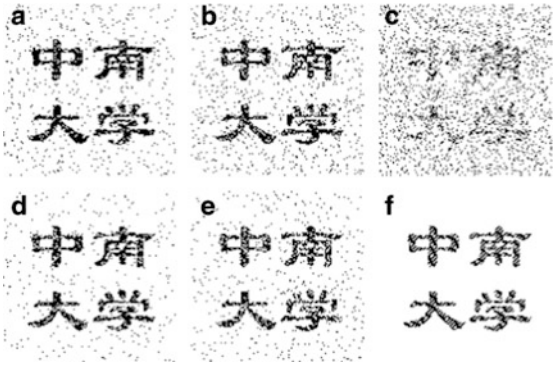

Fig. 8.21 Robust watermarks extracted from attacked host image (**a**) JPEG compress (QF = 50); (**b**) Add Gaussian white noise ($\sigma 2 = 24$); (**c**) Median filtering (radius = 2 pixel); (**d**) Blur (radius = 2 pixel, threshold = 10.0); (**e**) Sharping; (**f**) Cut a quarter of area

where HD(·,·) denotes the Hamming distance between two binary strings, i.e., the number of bits different in the two binary strings. Results show that the robust and fragile watermarks extracted from the watermarked image without any attacks are both with $NHS = 1$. To check the robustness and fragility of the algorithm, we perform several attacks on the watermarked image, including JPEG compression, adding channel additive white Gaussian noise (AWGN), image median filtering, blurring, sharpening, cropping are performed on the watermarked image. The extracted watermark $\mathbf{W}_R$ from watermarked image with these attacks are depicted in Fig. 8.21, the *NHS* values of $\mathbf{W}_R$ and $\mathbf{W}_F$ are listed in Tables 8.4 and 8.5, respectively.

The semi-fragile watermarking is robust to mild modification such as JPEG compression, channel noise and general mild image processes, but fragile to malicious attacks. In Fig. 8.22, we depict two malicious attacks, including image cropping and image content tampering. For each case, the tamper detection matrix can locate the attack position. The fragile watermark, therefore, can be used to verify the watermarked image.

If we consider the situation that the image is mildly modified such as severe AWGN while on the other hand it is maliciously attacked such as image content tampering attack. In our experiment, after a AWGN attack with $\sigma^2 = 18$, the watermarked image is maliciously attacked by adding word "Tamper" and changing the Lena's face into Elaine's face. Figure 8.23 shows the results of PP operation when using different parameters. It is very like the erosion operation in image processing. When $\alpha = 1$, $\beta = 3$, we can tell the malicious attack from the mild modification very clearly. This means we define one-order neighborhood, and set the threshold in the middle of total element number. The results of above experiments show that our algorithm is capable of tamper detection with relative precise localization.

Table 8.4 The NHS of robust watermark

	Fig. 8.3a	Fig. 8.3b	Fig. 8.3c	Fig. 8.3d	Fig. 8.3e	Fig. 8.3f
NHS($\mathbf{W}_R$)	0.950	0.925	0.821	0.959	0.944	0.970

Table 8.5 The NHS of semi-fragile watermark

	JPEG compress (QF = 50)	AWGN ($\sigma 2 = 24$)	Median filtering (radius = 2 pixel)	Blur (radius = 2 pixel, threshold =10)	Sharping	Cut a quarter of area
NHS (WR)	0.746	0.813	0.534	0.754	0.680	0.876

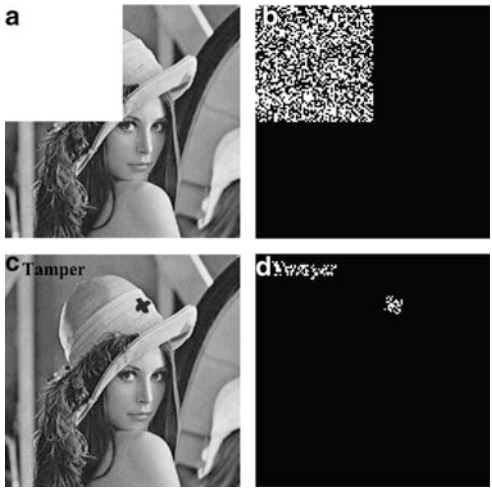

Fig. 8.22 Malicious attacks on watermarked images and the tamper detection matrix. (**a**) Image is cut a quarter of area in the upper-left corner; (**b**) Tamper detection matrix of (**a**); (**c**) Adding content tamper; (**d**) Tamper detection matrix of (**c**)

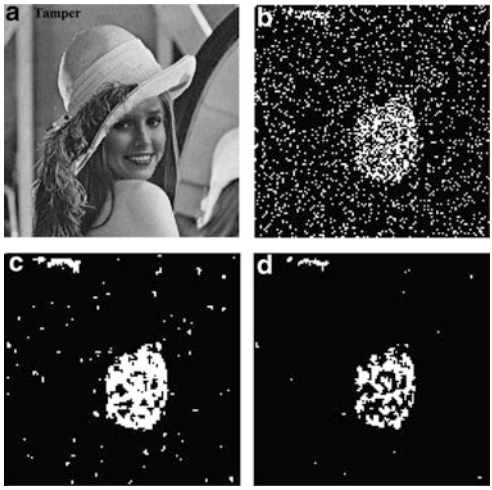

Fig. 8.23 Tampered watermarked image and pp operation on tamper detection matrix. (**a**) Tampered watermarked image; (**b**) No pp operation; (**c**) $\alpha = 1, \beta = 3$, (**d**) $\alpha = 1, \beta = 4$

8.3.4 Summary

An efficient multipurpose watermarking scheme based on DWT and chaotic map has been presented. It can not only achieve copyright protection purpose, but also achieve image authentication purpose. Experimental results demonstrate that the proposed scheme can be used for copyright protection by extracting the robust watermark, and it can also be used for image authentication by the tamper detection matrix of semi-fragile watermark. The main characteristic of this scheme is as follows. (1) the robust watermark for image copyright protection is robust to many mild and malicious attacks, while the fragile watermark for image authentication is robust to mild modification and fragile to malicious attack. (2) The fragile watermark has the ability of relative precise localization. (3) By employing optimization of watermark embedding method, enhanced invisibility.

8.4 Digital Video Watermarking Algorithm Based on Chaos and Motion Vector

As the new generation of video codec standard H.264 has widely used, more and more video using H.264 coding. Recently, the video copyright protection based on H.264 standard has attracted more attention [13–15]. Focused on the next-generation video compression standard H.264, a video watermarking scheme based on chaotic scrambling and the motion vector field is proposed. It belongs to a compressed domain video watermarking algorithm. Firstly, the watermarking information is scrambled in the motion vector field by a chaotic sequence. Secondly the scrambled watermark is embedded adaptively and extracted. Finally, the original watermarking information is recovered through scrambling inversion transformation. Pseudo-random and initial value sensitivity of chaotic sequence provides a guarantee for the security of the watermark information [16]. The results showed that both subjective and objective video quality is equivalent with the original video sequence, and bit rate error is small.

8.4.1 Image Scrambling Algorithm Based on Chaotic Sequence

Chaotic sequence is sensitive to initial conditions, and has good statistic property. Furthermore, it is generated easily. If it is used to scramble the watermark image, it has good effect, and will improve the security of the watermarking technology. So the application of chaotic sequence in digital watermark is a hot topic at present.

8.4.1.1 TD-ERCS Chaotic System

In 2004, Prof. Sheng et al. proposed a new discrete chaotic system, named TD-ERCS [17]. It has the properties of zero relativity and stable probability

distributing. In fact, it is a kind of iterative algorithm. Given two initial values x_0, tanα and two parameters μ、 m, and then we have

$$y_0 = \mu\sqrt{1 - {x_0}^2} \tag{8.23}$$

$${k_0}' = -\frac{x_0}{y_0}\mu^2 \tag{8.24}$$

$$k_0 = \frac{\tan\alpha + {k_0}'}{1 - {k_0}'\tan\alpha} \tag{8.25}$$

After computing the initial stats of TD-ERCS, other states can be calculated by

$$x_n = -\frac{2k_{n-1}y_{n-1} + x_{n-1}(\mu^2 - k_{n-1}^2)}{\mu^2 + {k_{n-1}}^2}, \quad n = 1, 2, 3, \ldots \tag{8.26}$$

$$k_n = \frac{2k'_{n-m} - k_{n-1} + k_{n-1}k'^2_{n-m}}{1 + 2k_{n-1}k'_{n-m} - k'^2_{n-m}}, \quad n = 1, 2, 3, \ldots \tag{8.27}$$

$$k'_n = -\frac{x_n}{y_n}\mu^2, \quad n = 1, 2, 3, \ldots \tag{8.28}$$

$$y_n = k_{n-1}(x_n - x_{n-1}) + y_{n-1}, \quad n = 1, 2, 3, \ldots \tag{8.29}$$

$$k'_{n-m} = \begin{cases} -\dfrac{x_{n-1}}{y_{n-1}}\mu^2, n<m \\ -\dfrac{x_{n-m}}{y_{n-m}}\mu^2, n \geq m \end{cases}, \quad n = 1, 2, 3, \ldots \tag{8.30}$$

Where $x_0 \in [-1, 1]$, tan$\alpha \in (-\infty, \infty)$, $\mu \in (0.05, 1)$, $m = 2, 3, \ldots$. n is the iteration number, and m is the tangent delay parameter. x_n and k_n are two independent and irrelevant chaotic sequences, which is produced by the formula above. Generally, x_0, α, μ and m are used as the secret keys. When $m = 1$, there exists a square attractor. When $m > 1$, the system is ergodic. Research shows that TD-ERCS has potential application value on cryptology [18,19].

Generation and Analysis of Chaotic Pseudo-Random Sequence

1. Chaotic pseudo-random sequence generator
   At present, most of generators of pseudo-random sequence are designed by an iteration formula [20]. A pseudo-random sequence with numbers enough is obtained by iterations. In this paper, generation steps of chaotic pseudo-random sequence are as follows.

(1) x_n and k_n are generated by TD-ERCS system, which are independent and non-related to each other. Then one of them is chosen as a using sequence $\{x_i\}$.
(2) The real number x_i is represented by b_n with p bits, where $\mathrm{n} = 1, 2, \ldots, p$.. The value of b_n is zero or one, i.e. $x_i = 0.b_1b_2\ldots b_p$. p means a kind of precision. The bigger p is, the higher the precision is. To chaotic pseudo-random sequence $\{x_i\}$, $\{\mathrm{s}_i\}$ is formed by choosing the kth bit binary number b_k.

2. Performances analysis of the chaotic pseudo-random sequence

(1) Balance analysis
For an ideal random binary sequence

$$s_n = \{s_1, s_2, \cdots, s_i, \cdots, s_n\}, \tag{8.31}$$

where n is the length of sequence. It should have a balance ratio of 0 to 1, i.e. $P\{s_i = 0\} = P\{s_i = 1\}$. It means the number of $\{s_i = 0\}$ and $\{s_i = 1\}$ is equal, and it satisfies the requirement of pseudo-random sequence.
(2) Correlation analysis
Correlation function of two pseudo-random sequences is defined as normalized cross- covariance function.

$$\begin{cases} C_{ij}(\tau) = \hat{C}_{ij}(\tau)\sqrt{\hat{C}_{ii}(0)\hat{C}_{jj}(0)} \\ \hat{C}_{ij}(\tau) = \dfrac{1}{N}\displaystyle\sum_{n=1}^{N-|\tau|}(b_{n,i} - b_{A,i})(b_{n+|\tau|,j} - b_{A,j}), \end{cases} \quad |\tau| = 1, \cdots, 2N-1 \tag{8.32}$$

where, C_{ij} means the cross-covariance between the ith pseudo-random sequence and the jth pseudo-random sequence. b_{nj} is the nth bit of the ith pseudo-random sequence, and b_{Ai} is the average value of b_{nj}.

Autocorrelation function of an ideal pseudo-random sequence is a δ function. Autocorrelation is a special form of cross-correlation, and represents by C_{ij}, and then we have

$$\begin{cases} C_{ii}(\tau) = \hat{C}_{ii}(\tau)/\hat{C}_{ii}(0) \\ \hat{C}_{ij}(\tau) = \dfrac{1}{N}\displaystyle\sum_{n=1}^{N-|\tau|}(b_{n,i} - b_{A,i})(b_{n+|\tau|,i} - b_{A,i}), \end{cases} \quad |\tau| = 1, \cdots, 2N-1 \tag{8.33}$$

Obviously, the auto-correlation of the chaotic pseudo-random sequence generated by TD-ERCS is similar to δ function, while its cross-correlation is close to zero. It satisfies the requirements of pseudo-random sequence [19].

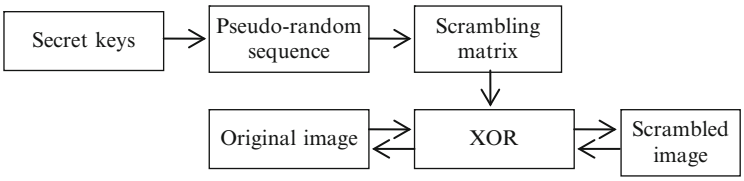

Fig. 8.24 Principle diagram of chaotic scrambling

Chaos Scrambling Algorithm for Binary Images

A binary image is a digital image with only two pixel values. So people call it B&W or single image. Binary image often storages in the form of bitmap, and it has small storage space, and fast processing speed. So many watermarking algorithms employ binary image as secure information to embed in a data carrier. Before it is embedded to a data carrier, the pre-treatment steps are needed firstly. During the process of pretreatment, scrambling approach is used to encrypt the watermarking information. In fact, scrambling images means encrypting images. If a scrambled image is taken as secure information to be hided in data carrier, it will improve the robustness of hiding carrier, and will not cause the data redundancy. To improve the security of watermark image, the chaotic pseudo-random sequence produced by TD-ERCS is used to scramble the binary watermark image.

1. Principle of scrambling algorithm
   The principle diagram of chaotic scrambling is presented in Fig. 8.24. Firstly, chaotic pseudo-random sequence is generated based on TD-ERCS system, and then the binarization and two-dimensional treatment are done to get the binary scrambling matrix. Finally, XOR operation is used between scrambling matrix and image matrix to obtain the scrambled image. To recover the original image, an inverse process is needed as shown in Fig. 8.24.

2. Design of image scrambling algorithm based on TD-ERCS

   The steps are as follows.

(1) Generating two sequences x_n and k_n by TD-ERCS system, and choosing one of them as a scrambling sequence $\{x_i\}$.
(2) Transform real number sequence to binary sequence b_n. For the chaotic pseudo-random sequence $\{x_i\}$, a new sequence $\{s_i\}$ is obtained by choosing the kth bit of the binary b_k.
(3) Transform one dimension sequence $\{s_i\}$ to two dimensions scrambling matrix $D(m, n)$. The row number m and the column n are corresponding to the row and column number of the image needed to be scrambled.
(4) XOR operation is made between matrix $D(m, n)$ and the original image matrix, and the scrambled image is obtained.

The steps above are used to the scrambled image, and after an inverse scrambling process, the original image can be recovered successfully.

Fig. 8.25 Scrambling effects of the watermark image. (**a**) The initial image; (**b**) the scrambled image; (**c**) the recovered image

Fig. 8.26 The images of scrambling inverse transformation with wrong keys. (**a**) key x0 with error 10–16; (**b**) key α with error 10–16; (**c**) key μ with error 10–16; (**d**) key m = 10

3. Experimental results and performance analyzing

(1) Analysis of sensitivity of secret keys

Four secret keys used in this algorithm are the four initial values of TD-ERCS, and are set to $x_0 = 0.7654$, $\alpha = 0.9876$, $\mu = 0.7123$ and $m = 9$ randomly. Because the chaotic system is very sensitive to its initial value, the scrambling algorithm in this chapter is also sensitive to the initial value. A tiny change of the initial values (secret keys) will cause the initial image to another scrambled image in disorder, which can not be recovered easily. The experimental results were presented in Figs. 8.25 and 8.26. In the experiment, the watermark image is a 40×40 binary image of "Central South University (in Chinese)." Figure 8.25 is the experiment of scrambling and scrambling inverse transformation. The inverse transformation of the scrambled image is the same with the scrambling of the original image. Figure 8.26 verified the sensitivity of the scrambling algorithm to the secret keys. For the scrambling inverse transformation, even though the error of secret keys is very tiny, there is no way to recover the initial image correctly.

The experiments results showed that the scrambling algorithm in this paper is very sensitive to the initial secret keys

(2) Analysis of the correlation between adjacent pixels

The effect of images scrambling is inversely proportional to the correlation of the adjacent pixels. For a binary image, the scrambling degree doesn't be measured

by the gray scale difference of an image. There are only two pixel values 1 or 0 for a binary image, so the pixel difference of an image is calculated by

$$GD_v(x,y) = \frac{|G(x,y) - G(x-1,y)| + |G(x,y) - G(x+1,y)|}{2}, \tag{8.34}$$

$$GD_h(x,y) = \frac{|G(x,y) - G(x,y-1)| + |G(x,y) - G(x,y+1)|}{2}, \tag{8.35}$$

where $GD_v(x, y)$ and $GD_h(x, y)$ are the pixels difference of an image on vertical direction and horizontal direction respectively. Suppose the size of an image is $M \times N$, then the vertical average adjacent pixels difference and the horizontal average adjacent pixels difference are represented by

$$E(GD_v(x,y)) = \frac{\sum_{x=2}^{M-1}\sum_{y=1}^{N} GD_v(x,y)}{(M-2)\times N}, \tag{8.36}$$

$$E(GD_h(x,y)) = \frac{\sum_{x=1}^{M}\sum_{y=2}^{N-1} GD_v(x,y)}{M\times(N-2)}. \tag{8.37}$$

The scrambling degree on vertical direction and horizontal direction is

$$GDD_v(I,I') = \frac{E'_v(GD(x,y)) - E_v(GD(x,y))}{E'_v(GD(x,y)) + E_v(GD(x,y))}, \tag{8.38}$$

$$GDD_h(I,I') = \frac{E'_h(GD(x,y)) - E_h(GD(x,y))}{E'_h(GD(x,y)) + E_h(GD(x,y))}, \tag{8.39}$$

where E and E' represent the average adjacent pixels difference between the original image and the scrambled image. Obviously, the range of the scrambling degree is (–1, 1). In practice, the range of the scrambling degree is (0, 1). Figure 8.27 shows the experiment results of two binary images. The size is 40 × 40. Table 8.6 is the results of the scrambling degree of the two binary images.

From Table 8.6, the difference of the average adjacent pixels before scrambled is small. It means that the correlation of adjacent pixels of the original image is big. The difference of the average adjacent pixels keeps at 0.5, which means that the pixels are distributed randomly, and the correlation between adjacent pixels is small. The values of the vertical scrambling degree and the horizontal scrambling degree are above 0.4, so the scrambling effect is very good.

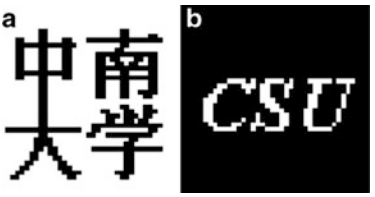

Fig. 8.27 Binary images for experiments

Table 8.6 Results of the scrambling degree of the two images

	Image (a)	Image (b)
$E(GD_v(x,y))$	0.1803	0.0697
$E(GD_h(x,y))$	0.1447	0.0737
$E'(GD_v(x,y))$	0.5161	0.5201
$E'(GD_h(x,y))$	0.4993	0.4888
$GDD_v(I,I')$	0.4823	0.7635
$GDD_h(I,I')$	0.5506	0.7380

8.4.2 Digital Video Watermarking Algorithm Based on Motion Vector

8.4.2.1 Video Decoding Standard H.264/AVC

Because of the outstanding performance of H.264 standard in compression, it plays an important role in various fields such as digital TV broadcasting, video, real-time communication and network video streaming media. Good network affinity is also an important feature of H.264. It is made good use in real-time video communications. Compression efficiency and transmission reliability are the focuses of H.264, which are different from MPEG-4. Thus, its application is very comprehensive, and the basic grade is used for "video conversation," while expansion grade is used for network video streaming quality, and the main grade is used for consumer electronics applications.

1. H.264/AVC encoder and decoder
   H.264 only provides syntax of encoded video bit stream and the corresponding decoding method. Specifically, how to implement codec operates flexibly. The various functional blocks of H.264 and other codec standards are almost the same, but it is different in the implementation details. Changing in video content leads to change in spatial detail, therefore we need appropriate adaptive technology to handle this situation. Because the channel is changing all the time in a harsh environment, the Internet sometimes is smooth, and sometimes is blocked. Specially, the wireless network is serious decline sometimes. The adaptive

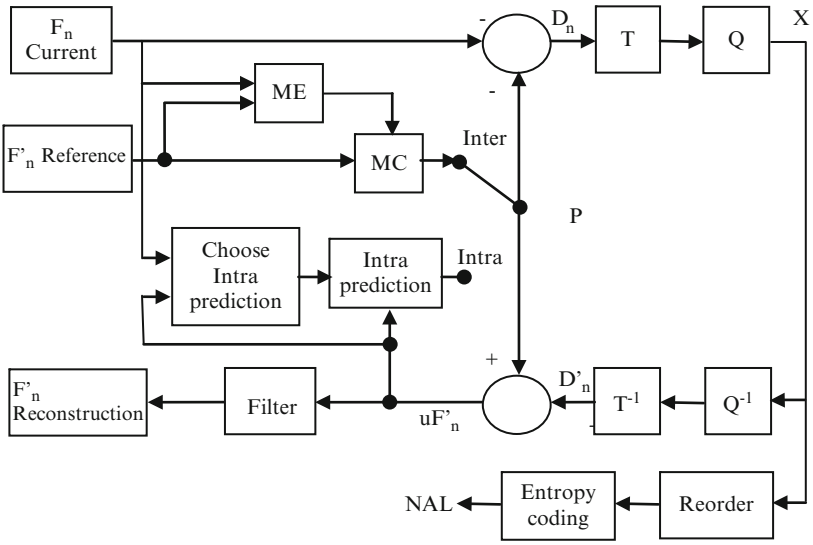

Fig. 8.28 H.264 encoder

technology greatly improves the compression performance of H.264. Next section, we introduce H.264 encoder and decoder and some of its key features briefly.

(1) H.264 encoder

The encoder uses hybrid coding method of transform and prediction. As shown in Fig. 8.28, the input frame or field is processed in the encoder based on macro-block F_n units. F'_{n-1} presents the reference image. First, it is processed by the method of intraframe or interframe prediction encoding. The intraframe prediction encoding is that its predictive value of PRED is obtained by the encoded reference image after employing motion compensation of current film. To improve forecast accuracy and the compression ratio, we select the actual image of reference in the frames which have been re-encoding and decoding and filtering in the past or the future (display orders). The predicted value of PRED minuses the current block, and generate a residual block Dn. After the block is transformed and quantified, we can get the transform coefficients X with entropy coding. Finally it generates a compressed stream after composited with the decoding information, and it is transmitted and deposited through the NAL.

As discussed above, in order to provide further prediction of the reference image, the encoder must have a function of image reconstruction. Therefore D′n obtained from the residual image through inverse quantify, inverse transform adds the predictive value *P* to produce uF′n (unfiltered frames). To eliminate the noise produced in encoding and decoding loop and improve the quality of the reference

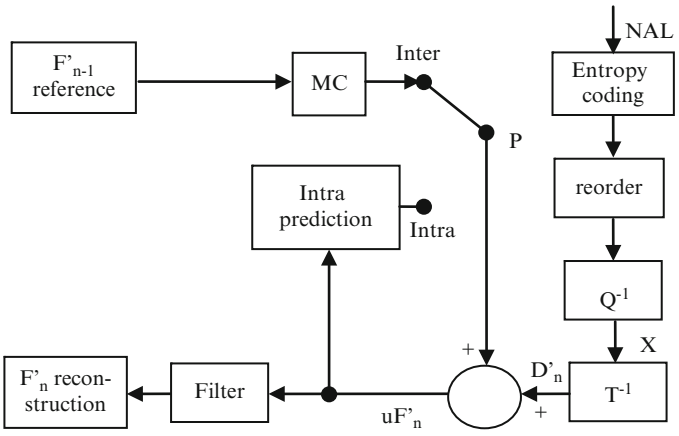

Fig. 8.29 H.264 decoder

frame image (i.e. improve the performance of compressed image), a loop filter is designed. The output F'n of the filter, called reconstruction image, is used as reference images.

(2) H.264 decoder

As shown in Fig. 8.28, a compressed H.264 bit stream is outputted in the NAL format at the encoder. As shown in Fig. 8.29, a set of quantified transform coefficients X is obtained after entropy coding, and residuals D′n is obtained after inverse quantify, inverse transform. By using the head information of the bit stream by decoding, the decoder generates a prediction block PRED, which is equivalent to the original forecast block PRED at the encoder. uF′n is obtained by adding PRED with D′n, and F'n is produced after uF'n is filtered. This F′n is the last decoding output image.

8.4.2.2 Choose the Watermark Embedding Location

There are three kinds of frames I, P, B in H.264 video. I frame and P frame should be used as a reference frame of the follow-up motion estimation, and B frame isn't used as a reference frame, so we chose to embed the watermark information in the B frames, which will not accumulate error caused by the follow-up frames. According to HVS human vision model, eyes are not sensitive to the area with movement, which manifested in the larger value of the motion vector area, so in this algorithm we select the largest block of motion vectors to embed watermark in each macro block. In this algorithm, the watermark is embedded by modifying the parity. When embedding the block motion vector, we need to determine the size of a motion vector. But the error accumulation after embedding watermark in front of the block will affect the follow-up block motion estimation, and it has a greater influence on

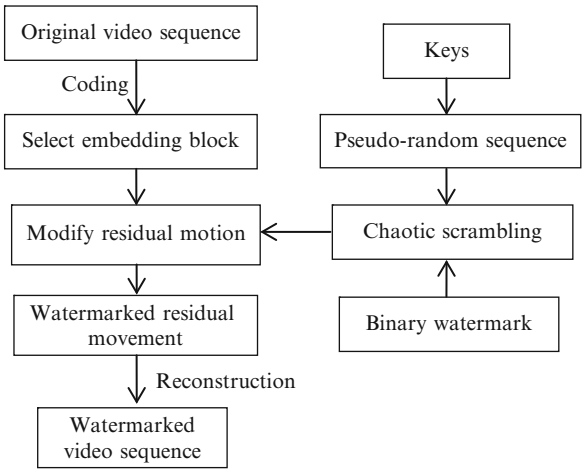

Fig. 8.30 Algorithm diagram of video watermark embedding

the extracted watermark. While there does not exist this problem for the residual motion. What transmitted in the code stream in H.264 are sport residuals. As long as finding the correct embedding point, the value of sport residual is right.

8.4.2.3 Watermark Embedding Algorithm

The algorithm diagram of video watermark embedding is shown in Fig. 8.30, and its steps are as follows.

(1) In the encoder, select the B frame video frame, and cheek all the blocks within a macro block. Then choose amplitude of the largest motion vector block as the block in which a watermark embed, and denoted it by $block(i_m, \mathrm{j}_m)$, where i_m, j_m represent horizontal and vertical coordinates in 4×4 blocks, respectively.
(2) Compare the horizontal component mv_{h} and vertical component mv_{v} of the current motion vector block, which a watermark will be embedded in, and to find out the larger amplitude. If the larger value is not zero, the watermark is embedded, otherwise skip the current macro block.
(3) Scrambling binary watermarking by the chaotic sequence produced by TD-ERCS system.
(4) Modify the parity of movement residuals mvd_{n} or mvd_{v}.

Embedding rules are as follows:

(1) When $|mv_k| >= |mv_v|$, the watermark information is embedded in horizontal component mvd_h of the movement residuals. Then its parity is modified. The rules are as follows:

$$\begin{cases} wm_k = 0 \; and \; |mvd_h|\%2 = 1 \\ wm_k = 1 \; and |mvd_h|\%2 = 0 \end{cases} \quad mvd_h = \begin{cases} mvd_h - 1, & mv_h < 0 \\ mvd_h + 1, & mv_h \geq 0 \end{cases} \tag{8.40}$$

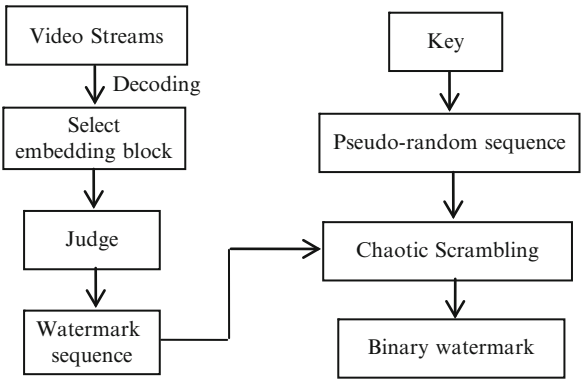

Fig. 8.31 Algorithm diagram of watermarking extraction

(2) When $|mv_k| < |mv_v|$, the watermark information is embedded in vertical component mvd_h of the movement residuals, and its parity is modified. The rules are as follows:

$$\begin{cases} wm_k = 0 \; and |mvd_v| \%2 = 1 \\ wm_k = 1 \; and |mvd_v| \%2 = 0 \end{cases} \quad mvd_v = \begin{cases} mvd_v - 1, & mv_v < 0 \\ mvd_v + 1, & mv_v \geq 0 \end{cases} \tag{8.41}$$

Where wm_k is the k place for the watermark sequence in (8.40) and (8.41). mv_k is the horizontal component and mv_v is the vertical motion vector components. mvd_h is the residual horizontal component and mvd_v is the residual vertical movement components. So motion vector is

$$mv = mvd + pred\_mv, \tag{8.42}$$

where *pred_mv* is the motion vector prediction value. *mvd* is the motion residuals. When *mv* is less than 0, minus the motion residual by 1, and plus 1 vice versa. To determine the embedded point, one need to compare the absolute value of motion vector between horizontal component and vertical component, and the motion vector is the sum of the predicted values and residuals. The motion vector is decreasing as the residuals decreasing. When motion vector becomes negative, its absolute value is larger, and the same vice versa. It can reduce the error rate of determining an embedded point.

8.4.2.4 Watermark Extraction Algorithm

The block diagram of the watermark extraction algorithm is shown in Fig. 8.31. The process of watermark extracting is the reverse process of watermark embedding. It does not need the original video binary image, so it belongs to blind extraction. At the decoder, decoding the residual motion, motion vector, and checking all the blocks of macro block. After we find the maximum amplitude of the block motion vector, we compare the amplitude of the horizontal component and vertical

Table 8.7 JM operation parameters

Profile	Baseline
Y:U:V	4:2:0
Image format	CIF
Number of reference frames	5
Motion search method	Full search
QP	28

Fig. 8.32 Binary image watermark

component of the motion vector in this block. We should select a larger value to extract the watermark, and determine the parity of the sub residual value at this movement. If it is odd, then the watermark bit is "1." If it is even, then the watermark bit is "0." All the scrambled watermark information can be recovered by employing this method. Finally, chaotic scrambling inverse transform is applied, and the binary watermark image is recovered correctly.

8.4.2.5 Experiment Results and Analysis

In this experiment, watermark embedding and extraction experiments are on the test platform, named JM86 H.264, and the main parameters of JM86 test platform are listed in Table 8.7. The performance analysis was done for the video watermarking algorithm. Three videos, including foreman sequence, bus sequence and mobile sequences, are used as test video sequence. A binary image shown in Fig. 8.32 is taken as the watermark.

(1) Video quality test before and after the watermark embedding

To analyze the effect of the watermarked video quality, two parameters are introduced, *i.e*. the subjective visual and *PSNR* (Peak Signal-to-Noise ratio) values. Table 8.3 presents *PSNR* values before and after the watermarked video sequence. By comparing, we found that the quality of the video is almost not changed, and there is no obvious distortion of the video. So the embedded watermark has tiny impact on video quality. Figure 8.33 gives the comparison of a frame before and after the video embedded image.

Furthermore, Table 8.8 shows that *NC* (Normalized Correlation) values of the extracted watermark image by three experimental video sequences and original watermark image, which are all close to 1. Figure 8.34 shows the corresponding extracted binary watermark image. Obviously the watermark can be extracted correctly.

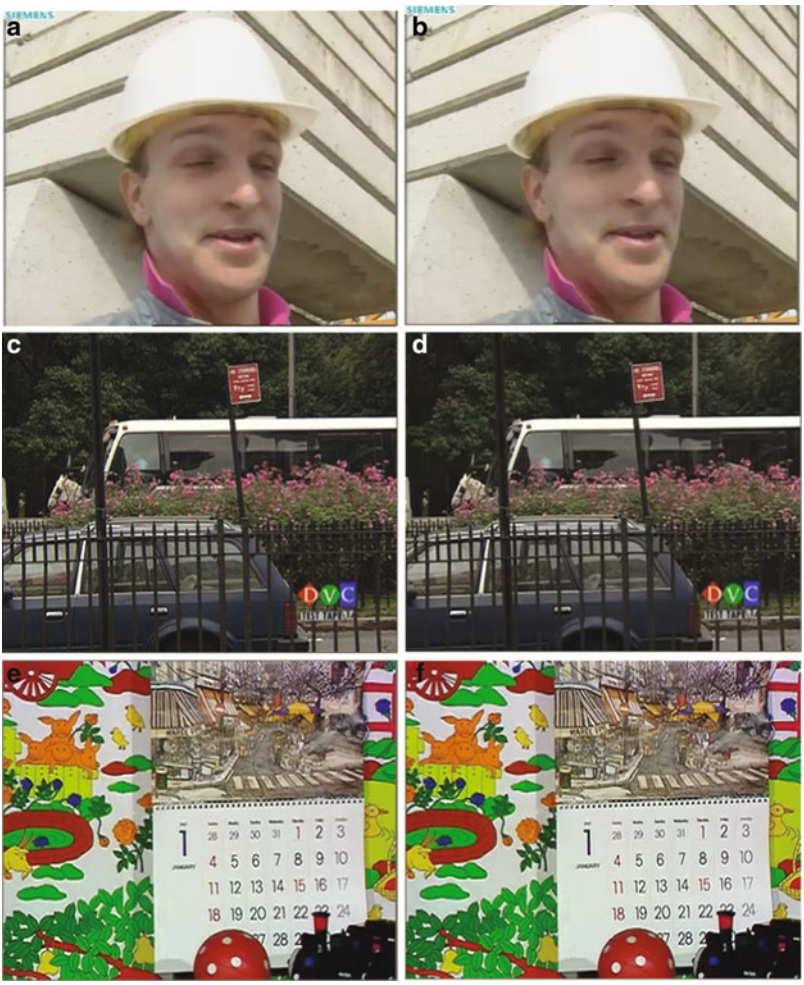

Fig. 8.33 Video image effect before and after the watermark embedded (**a**) the original foreman video frame; (**b**) the foreman video frame of watermark embedded; (**c**) the original bus video frame; (**d**) the bus video frame of watermark embedded; (**e**) the original mobile video frame; (**f**) the mobile video frame of watermark embedded

Table 8.8 The values of video sequences before and after the watermark and NC value of watermarking extraction

Video sequence	PSNR of original video sequence			PSNR of watermarked video sequences			NC
	Y	**U**	**V**	**Y′**	**U′**	**V′**	
Foreman	37.15	41.16	44.03	37.15	41.16	44.03	0.9917
Bus	34.72	40.21	41.90	34.73	40.21	41.89	0.9976
Mobile	34.07	36.92	36.57	34.07	36.92	36.57	0.9637

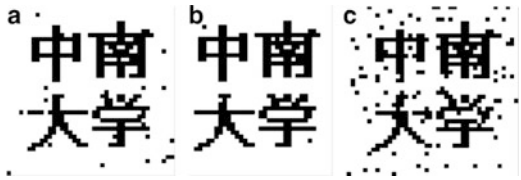

Fig. 8.34 Binary watermark image extracted by three groups of the experimental (**a**) the watermark extracted from foreman video sequence (**b**) the watermark extracted from bus video sequence (**c**) the watermark extracted from mobile video sequence

Table 8.9 Bit rate changes before and after watermark embedding

Test sequence	Before watermark embedding	After watermark embedded	Difference of bit rate
Foreman	525.81	526.30	0.09 %
Bus	1348.47	1348.91	0.03 %
Mobile	1778.39	1778.85	0.03 %

Table 8.10 Experimental data of frame deleting attack

Frame deletion rate	NC
9 %	0.9300
11 %	0.9185
14 %	0.9051
17 %	0.8992
20 %	0.8816
23 %	0.8731
26 %	0.8635
28 %	0.8497

(2) The test of bit rate changes

Video watermarking algorithms require the effect of bit-rate after the watermark embedding to be very small. Table 8.9 presents the result of the impact of embedding to video bit rate. The change rate is very small, which is within 0.1%.

(3) Attacking test

Frame deleting and frame inserting are used as the attack test in this algorithm. When the frame deletion rate is less than 25 %, and *NC* value is more than 0.85, which are listed in Table 8.10. In these cases, the watermark can be extracted correctly. It means that the algorithm has a good robustness to frame deletion attack.

Table 8.11 shows the experimental data of frame inserting attack. From the table, we found that when the frame insertion rate is less than 23 % in the experiments, and *NC* value is more than 0.85. So the watermark can be extracted accurately. It shows that the watermarking algorithm has a good resistance to frame insertion attacks.

Table 8.11 Experimental data of frame inserting attack

Frame insertion rate	NC
3 %	0.9598
6 %	0.9443
9 %	0.9296
11 %	0.9177
14 %	0.8966
17 %	0.8804
20 %	0.8639
23 %	0.8577

(4) Safety analysis

Using the method of images scrambling mentioned in Sect. 8.1 to preprocess the binary watermark image, which is equivalent of encryption the watermarking image before the watermark embedded, and then decrypted when extracting watermark. So even if the attacker access to the illegal watermark information from the video sequence, but they do not obtain the complete parameters of TD-ERCS chaotic system, they also can not recover the right watermark information. Due to the large key space of the TD-ERCS in the scrambling algorithm, the security of watermarking algorithm has been improved.

8.4.3 Summary

By making use of the excellent properties of the TD-ERCS chaotic system, the chaotic pseudo-random sequence is generated and is applied to scrambling the binary watermark image. A scrambling algorithm based on TD-ERCS chaotic system was proposed. Experiments results show the scrambling effect and security is improved, and it is applied for the pretreatment process of the watermark image to improve its security.

Based on the video coding standard H.264/AVC codec framework, and H.264 video encoding technology, a video watermarking scheme based on chaotic scrambling and the motion vector field was proposed, named a compressed domain video watermarking algorithm. The watermark is embedded adaptively in the motion vector field. The watermark information scrambled by chaotic sequence is extracted in the encoder, and then it is recovered to the original watermark through scrambling inversion transformation. Experimental results show that chaotic scrambling provides a security guarantees to the watermark embedding algorithm. At the same time, the watermark has good invisibility and robustness, and the increment of watermark-embedded video stream is small, which all are less than 0.1 %.

Acknowledgments This work was supported by the National Nature Science Foundation of People's Republic of China (Grant NO. 61073187), and the SRF for ROCS, SEM. The author Prof. Sun would like to express thanks to his graduate student Mr. Cheng Wei for his experiment research.

References

1. Dawei, Z., Guanrong, C., Wenbo, L.: A chaos-based robust wavelet-domain watermarking algorithm [J]. Chaos Solitons Fractals **22**(1), 47–54 (2004)
2. Schmitz, R.: Use of chaotic dynamical systems in cryptography [J]. J. Franklin. Inst. **338**, 429–41 (2001)
3. Shih, F.Y., Wu, S.Y.T.: Combinational image watermarking in the spatial and frequency domains[J]. Pattern Recogn. **36**(4), 969–975 (2003)
4. Hwang, M.S., Chang, C.C., Hwang, K.F.: A watermarking technique based on one-way hash functiojn[J]. IEEE Trans. Consum. Electron. **45**(2), 286–294 (1999)
5. Qi, S., Zhu, G., Rong, T., et al.: A kind of digital watermarking algorithm based on modulo[J]. Acta Electron. Sinica **30**(6), 890–892 (2002)
6. Arnold, V., Avez, A.: Ergodic Problems in Classical Mechanics [M]. Benjamin, New York (1968)
7. Paquet, A.H., Ward, R.K., Pitas, I.: Wavelet pacets-based digital watermarking for image verifica tion and authentication [J]. Signal Process. **83**(10), 2117–2132 (2003)
8. Liu, S., Yao, H., Gao, W., et al.: An image fragile watermark scheme based on chaotic image pattern and pixel-pairs [J]. Appl. Math. Comput. **185**(2), 869–882 (2007)
9. Zhu, B.B., Swanson, M.D., Tewfik, A.H.: When seeing is not believing [J]. IEEE Signal Process. Mag. **21**(2), 40–49 (2004)
10. Yuan, H., Zhang, X.P.: A multiscale fragile watermarking based on the Gaussian mixture model in the wavelet domain. Proceedings of the 2004 International Conference on Acoustics, Speech and Signal Processing, vol. 3, pp. 413–416, Montreal, QC, Canada, May 2004
11. Ding, K., He, C., Jiang, L.G., Wang, H.X.: Wavelet-based semi-fragile watermarking with tamper detection. Ieice Transactions on Fundamentals, vol. E88–A, no. 3, pp. 787–790, 2005
12. Lu, Z., Xu, D., Sun, S.: Multipurpose image watermarking algorithm based on multistage vector quantization [J]. IEEE Trans. Image Process. **14**(6), 822–831 (2005)
13. Liu, Y., Zhao, J.: A new video watermarking algorithm based on 1D DFT and Radon transform. Signal Process. **90**(2), 626–639 (2010)
14. Kim, D.-W., Choi, Y.-G., Kim, H.-S., Yoo, J.-S., Choi, H.-J., Seo, Y.-H.: The problems in digital watermarking into intra-frames of H.264/AVC. Image Vision Comput. **28**(8), 1220–1228 (2010)
15. Yang, G., Li, J., He, Y., Kang, Z.: An information hiding algorithm based on intra-prediction modes and matrix coding for H.264/AVC video stream. Int. J. Electron. Comm. **65**(4), 331–337 (2011)
16. Singh, N., Sinha, A.: A chaotic system based fragile watermarking scheme for image tamper detection [J]. Int. J. Electron. Commun. **65**, 840–847 (2011)
17. Sheng, L., Sun, K., Li, C.: Study of a discrete chaotic system based on tangent-delay for elliptic reflecting cavity and its properties [J]. Acta Phys. Sinica **53**(9), 2871–2876 (2004)
18. Sheng, L., Cao, L., Sun, K., Wen, J.: Pseudo-random number generator based on TD-ERCS chaos and its statistic characteristics analysis[J]. Acta Phys. Sinica **54**(9), 4031–4037 (2005)
19. Sun, K., Tang, G., Sheng, L.: The complexity analysis of TD-ERCS discrete chaotic pseudo-random sequences [J]. Acta Phys. Sinica **57**(6), 3359–3366 (2008)
20. Fan, M., Wang, H.: Chaos-based discrete fractional Sine transform domain audio watermarking scheme [J]. Comput. Electr. Eng. **35**, 506–516 (2009)

Chapter 9
A New Direction in Utilization of Chaotic Fractal Functions for Cryptosystems

M.R.K. Ariffin, N.M.G. Al-Saidi, M.R.M. Said, Z. Mahad, and M.A. Daud

9.1 Introduction

Since 1989, many cryptosystems based on chaotic maps have been proposed for cryptographic implementation. These cryptosystems were motivated by the fact that chaotic maps are usually noise-like and very sensitive to initial conditions. Their sensitivity to initial conditions and their spreading out of trajectories over the whole interval seems to be a model that satisfies the classic Shannon requirements of confusion and diffusion.

In 1998 Baptista proposed a new cryptosystem based on the ergodic property of chaotic systems [4]. Since then a number of new algorithms based on variations of Baptista's have been published. It encrypts the message text into the number of iterations needed for a chaotic map to reach a region on a phase space (i.e. a particular section on the range of the map) that corresponds to that text. He demonstrated his approach using a simple one-dimensional logistic map given by the following equation: $x_{n+1} = bx_n(1 - x_n)$ where b is the gain and $x_n \in (0, 1)$. Since the ciphertexts are small integers, they are suitable to be transmitted through today's public digital networks. In Baptista's original work, in order to avoid statistical and differential cryptanalysis, a random number is generated each time the chaotic trajectory has reached the desired region. If it is greater than a threshold η, the current number of iterations will be transmitted as the ciphertext. Otherwise, the iteration will continue. An interesting feature that could be observed through this cryptosystem is its ability to produce different ciphertexts that correspond to the same plaintext. This feature would result in an attacker having different ciphertexts that could correspond either to the same or different plaintext. This unique feature was once the source of

M.R.K. Ariffin (✉) • N.M.G. Al-Saidi • M.R.M. Said • Z. Mahad • M.A. Daud
Al-Kindi Cryptography Research Laboratory, Institute for Mathematical Research, Universiti Putra Malaysia, 43400 Serdang, Selangr, Malaysia
e-mail: rezal@math.upm.edu.my; nadiaalsaidi@hotmail.com; rushdan@math.upm.edu.my; zahari@inspem.upm.edu.my; azlan@inspem.upm.edu.my

S. Banerjee et al. (eds.), *Applications of Chaos and Nonlinear Dynamics in Science and Engineering - Vol. 2*, Understanding Complex Systems,
DOI 10.1007/978-3-642-29329-0_9,

strength for Vigenere's cipher. Further analysis conducted on this cryptosystem revealed that it also inherits the same weakness that Vigenere's cipher suffered from. Namely it was the re-usage of the key. This resulted in the success of Alvarez's attack on it in 2003, which is known as the One Time Pad Attack [1]. In 2008, Ariffin attempted to modify the cryptosystem via an example in order for it to remain secure [2]. In 2009, Rhouma identified a flaw in the example Ariffin put forward [6]. In between the above mentioned works by in 2008 and 2009, Ariffin produced a theoretical explanation in order for the cryptosystem to remain secure which is independent of the example and was published in 2010 [3]. However, the work in 2010 by Ariffin still cited the flawed example due to the early submission of the article before Rhouma's work was published.

A fractal set is a set that fits a physical world model better than regular arrangements involving smooth curves and surfaces. Iterated function systems (IFS) provide a convenient framework for the description, classification, and communication of fractals. It has been used to define fractals. Due to their complicated mathematical structure and deterministic nature, especially their recursive construction, it has many applications in physics, chemistry, biology, engineering, and recently in cryptographic systems. In this article a fractal map is discretized in order for it to be suitable for cryptographic applications. The discretized map will be utilized to enhance a cryptosystem utilizing the Baptista mechanism. Each resulting consecutive ciphertext from the Baptista encryption technique will be paired to represent a coordinate on the xy-plane. It will be then operated with the discretized fractal map. The result will be resistant towards Alvarez's one time pad attack.

Mathematical and empirical analysis are conducted to determine the security of our suggested cryptosystem. Careful considerations are conducted during the design of this cryptosystem such that weaknesses that arise from the Baptista design will not occur. Mechanisms explaining the design of this new cryptosystem such that it does not inherit weaknesses from the previous Baptista design are also detailed out.

9.2 Background for Fractal and Iterated Function Systems (IFS)

The theory of fractal sets is a modern domain of research. Iterated function systems have been used to define fractals. Such systems consist of sets of equations, which represent a rotation, a translation, and a scaling. These equations can generate complicated fractal images.

Given a metric space (X, d), the space of all nonempty compact subset of X is called the Hausdorff space $H(X)$. The Hausdorff distance h is defined on $H(X)$ by,

$$h(P,Q) = max\{inf\{\varepsilon>0; Q \subset N\varepsilon(P)\}, inf\{\varepsilon>0; P \subset N\varepsilon(Q)\}\}, \tag{9.1}$$

Definition 1. *For any two metric spaces (X, d_x) and (Y, d_y), a transformation $\beta : X \to Y$ is said to be a contraction if and only if there exists a real number s, $0 \leq s \leq 1$, such that $d_y(\beta(x_i),\beta(x_j)) < sd_x(x_i, x_j)$, for any $x_i, x_j \in X$, where s is the* contractivity factor *for β.*

The following theorem, known as the *contraction mapping theorem*, states an important property of contractive transformations of a complete metric space within itself.

Theorem 1. *Let $\beta : X \to Y$ be a contraction on a complete metric space (X, d). Then, there exists a unique point $x_f \in X$ such that $\beta(x_f) = x_f$. Furthermore, for any $x \in X$, we have $\lim_{n\to\infty} \beta^{o n}(x) = x_f$, where $\beta^{o n}$ denotes the n-fold composition of β.*

A fractal is constructed from a collage of transformed copies of itself. It is inherently self-similar and infinitely scalable. The transformation is performed by a set of affine maps. An affine mapping of the plane is a combination of a rotation, scaling, a sheer and a translation in $\mathbb{R}^2$.

Definition 2. *Any affine transformation $\beta : \mathbb{R}^2 \to \mathbb{R}^2$ of the plane has the form,*

$$\begin{pmatrix} u \\ v \end{pmatrix} = \beta \begin{bmatrix} x \\ y \end{bmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix} = A\overrightarrow{X} + b \tag{9.2}$$

where $(u, v), (x, y) \in \mathbb{R}^2$, are any point on a plane.

By considering a metric space (X, d) and a finite set of contractive transformation $\beta_n : X \to X$, $1 \leq n \leq N$, with respective contractivity factors s_n, we proceed to define a transformation $B : H(X) \to H(X)$, where $H(X)$ is the collection of non-empty, compact subsets of X, by:

$$A = B(A) = \bigcup_{i=1}^{N} \beta_i(Q) \text{ for any } Q \in H(X) \tag{9.3}$$

It is easily shown that B is a contraction, with contractivity factor

$$s = \max_{1<n<N} sn$$

The mapping B is usually referred to as the Hutchinson operator. It follows from the contraction mapping theorem that, if (X, d) is complete, B has a unique fixed point $A \in H(X)$, satisfying the remarkable self covering condition.

$$A = B(A) = \bigcup_{i=1}^{N} \beta_i(A) \tag{9.4}$$

Definition 3. *A hyperbolic IFS $\{X;\beta_1,\beta_2,\ldots,\beta_n\}$ consists of a complete metric space (X, d) and a finite set of contractive transformation β_n: $X \to X$ with contractivity factors s_n, for $n = 1,\ldots,N$. The contractivity factor for the IFS is the maximum s among $s_1,\ldots,s_N$. The attractor of the IFS is the unique fixed point in H(X) of the transformation B defined by (9.3).*

However, given a set M, the question is how to find a contractive transformation B such that its attractor A is close to M? To answer this question we have to apply to the Collage Theorem.

Theorem 2. *For a set M and a contraction B with attractor A:*

$$h(M,A) \leq \frac{h(M,B(M))}{1-s} \tag{9.5}$$

where h is the Hausdorff Distance.

That is to say that M and A will be sufficiently close, if M and $B(M)$ are made close enough in terms of β_i, and combining the following two expressions:

$$B(M) = M, B(M) = \bigcup_{i=1}^{N} \beta_i(M)$$

which implies

$$\bigcup_{i=1}^{N} \beta_i(A) \approx M \tag{9.6}$$

So, M can be partitioned as: $M = \bigcup_{i=1}^{N} m_i$ and m_i can be closely approximated by applying a contractive affine transformation β_i on the whole M, where,

$$m_i = \beta_i(M) \tag{9.7}$$

The theory of IFS was extended to local IFS where each part of the image is approximated by applying a contractive affine transformation on another part of the image:

$$m_i = \beta_i(D_i) \tag{9.8}$$

where D_i is the bigger part from which m_i is approximate.

9.3 The Relationship Between Fractal, Chaos and Cryptography

Chaos theory has a close tie-in with fractals. Most of the attractors produced by chaotic dynamical systems are fractal sets. For example, the Lorenz attractor is a fractal of Hausdorff dimension equal to 2. 073. The chaotic behavior of a fractal is used to encrypt data, in fractal cryptography there are no rounds, instead iterations are used, and the security is based in the non-determinism of a recursive function (for calculating the nth iteration of a function you need to calculate $(n-1)$ iteration first).

In the literature, fractals theory has proved to be suitable in many fields and particularly interesting in various applications of image processing. First important

advances are due to M.F. Barnsley, who introduced for the first time the term of Iterated Function Systems (IFS) based on the self-similarity of fractal sets [5]. Barnsley's work assumes that many objects can be closely approximated by self-similarity objects that might be generated by use of IFS simple transformations. From this assumption, the IFS can be seen as a relationship between the whole image and its parts, thus exploiting the similarities that exist between an image and its smaller parts.

For our purposes we will consider an IFS consisting of the maps,

$$w_i(x,y) = \begin{pmatrix} a_i & b_i \\ c_i & d_i \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e_i \\ f_i \end{pmatrix}, i = 1, 2, \ldots, N$$

for $i = 1$. That is,

$$w_1 = \begin{pmatrix} x_{i+1} \\ y_{i+1} \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x_i \\ y_i \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix},$$

and we will let the matrix

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

consist of only elements within the set {0, 1}.
Next, the 2×1 matrix

$$B = \begin{pmatrix} x_i \\ y_i \end{pmatrix}$$

will consist of Baptista cipherrtext values, and the matrix

$$C = \begin{pmatrix} e \\ f \end{pmatrix}$$

will be equal to zero (i.e. $C = 0$).

9.4 Result: The Modified Baptista Type Chaotic Cryptosystem via Matrix Secret Key Based on IFS

9.4.1 Encryption

9.4.1.1 Preparing a Chaotic Map

- Assume that we construct a look-up table consisting of j ε-intervals.
- Represent each site with S_1, S_2, S_3. . S_j.

- The minimum value of the first interval is 0, and the upper bound of the interval is 1.
- Choose a one-dimensional chaotic map with the onto property. (In our article we will utilize the logistic map).

9.4.1.2 Preparing the Matrix Secret Key

- Generate a $k \times k$ matrix ($[A]_{k\times k}$) such that its inverse ($[A]^{-1}_{k\times k}$) exists.

$$A = \begin{pmatrix} M_{11} & M_{12} & \cdots & \cdots & M_{1+k} \\ M_{21} & \cdots & \cdots & \cdots & M_{2+k} \\ \vdots & \cdots & \cdots & \cdots & \vdots \\ \vdots & \cdots & \cdots & \cdots & \vdots \\ M_{k1} & M_{k2} & \cdots & \cdots & M_{kk} \end{pmatrix},$$

The matrix will consist elements only from the set $\{0, 1\}$. This matrix will be the secret key.

9.4.1.3 Preparing Distorted Plaintext

- Encrypt each plaintext via the Baptista method.
- The iteration numbers are denoted as C_1.
- Group each element of C_1 into matrix of dimension $k \times 1$. Then do the following matrix multiplication.

$$[C_2]_{k\times 1} = [A]_{k\times k} \times [C_1]_{k\times 1}$$

- $[C_2]_{k\times 1}$ is the ciphertext to be transmitted to the recipient.

9.4.2 Decryption

9.4.2.1 Multiply $[A]^{-1}_{k\times k}$ with the Following Ciphertexts ($[C_2]_{k\times 1}$)

- Do the following matrix multiplication:

$$[C_1]_{k\times 1} = [A]^{-1}_{k\times k} \times [C_2]_{k\times 1}$$

- Would result in a list of integers.
- Use each integer to iterate the logistic map. Start iterating the logistic map until it falls in the corresponding phase space of the first character and continue iterating until the final character to get the original plaintext.

9.5 Example

Let us use a 4-symbol source, $S_4 = \{s_1, s_2, s_3, s_4\}$. For illustrative purposes the key $x_0 = 0.232323$ and parameter $b = 4$ is utilized. The plaintext is given by $P = s_4, s_3, s_1, s_1, s_2, s_4, s_3, s_1$.

Site	Associated interval (phase space)
s_1	[0, 0.25)
s_2	[0.25, 0.5)
s_3	[0.5, 0.75)
s_4	[0.75, 1]

9.5.1 Encryption

- Choose $k = 2$.
- Preparing matrix key, let $A = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$,
- For each character P, was encrypt via Baptista cryptography method.
- We have the following ciphertext, C_1 : 2, 10, 2, 11, 5, 1, 2, 2.
- Do the following matrix multiplication $[C_2]_{2\times 1} = [A]_{2\times 2} \times [C_1]_{2\times 1}$.
- We pair two consecutive values to be points on the Cartesian plane

$$(x_1, y_1) = (2, 10), (x_2, y_2) = (2, 11), (x_3, y_3) = (5, 1), (x_4, y_4) = (2, 2).$$

- Do the matrix multiplication procedure.

$$\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 2 \\ 10 \end{pmatrix} = \begin{pmatrix} 12 \\ 10 \end{pmatrix}, \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 2 \\ 11 \end{pmatrix} = \begin{pmatrix} 13 \\ 11 \end{pmatrix}, \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 5 \\ 1 \end{pmatrix}$$
$$= \begin{pmatrix} 6 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 2 \\ 2 \end{pmatrix} = \begin{pmatrix} 4 \\ 2 \end{pmatrix}.$$

- We have the following ciphertext, C_2 : 12, 10, 13, 11, 6, 1, 4, 2.
- We transmit C_2 to recipient.

9.5.2 Decryption

- To decrypt we will use the inverse of A.

$$A^{-1} = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}$$

- We have the following ciphertext, C_2 : 12, 10, 13, 11, 6, 1, 4, 2.
- Do the following matrix multiplication $[C_1]_{2\times 1} = [A]^{-1}_{2\times 2} \times [C_2]_{2\times 1}$.
- We pair two consecutive values to be points on the Cartesian plane

$$(x_1^*, y_1^*) = (12, 10), (x_1^*, y_1^*) = (13, 11), (x_1^*, y_1^*) = (6, 1), (x_1^*, y_1^*) = (4, 2).$$

- Do the inverse matrix multiplication procedure.

$$\begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 12 \\ 10 \end{pmatrix} = \begin{pmatrix} 2 \\ 10 \end{pmatrix}, \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 13 \\ 11 \end{pmatrix} = \begin{pmatrix} 2 \\ 11 \end{pmatrix}, \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 6 \\ 1 \end{pmatrix}$$
$$= \begin{pmatrix} 5 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 4 \\ 2 \end{pmatrix} = \begin{pmatrix} 2 \\ 2 \end{pmatrix}.$$

- From the inverse matrix multiplication, we have the following ciphertext, C_1 : 2, 10, 2, 11, 5, 1, 2, 2.
- Iterate the chaotic map first 2-times, second 10-times and so on.
- Get the plaintext $P = S_4, S_3, S_1, S_1, S_2, S_4, S_3, S_1$.

9.6 Cryptanalysis Using Alvarez's One Time Pad Attack (Chosen Plaintext Attack)

In this section we will conduct an attack using the one-time pad attack designed by Alvarez in 2003 [1]. The matrix key we will use is given by:

$$A = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix},$$

- Start by requesting the ciphertext for S_1. After the procedure we will get $S_1^* = (8, 3, 2, 1, 4, 3, 12, 1, 2, 1, 6, 5,)$.
- For S_2 we have $S_2^*(15, 9, 4, 2, 4, 2, 16, 9, 16, 9, 10, 8, \ldots)$.
- For S_3 we have $S_3^* = (3, 2, 29, 20, 12, 2, 9, 5, 8, 2, 12, 2, \ldots)$.
- For S_4 we have $S_4^* = (4, 2, 9, 6, 5, 2, 4, 2, 9, 7, 7, 5, 5, 3, \ldots)$.
- With only the above information we will construct a partial one-time pad:

$$O = xxxs_3s_4s_3s_4xs_1xxs_1xs_1s_1\underline{s_2s_4}xxs_1xx\underline{s_1s_4}xxs_2xs_4x\underline{s_2s_4}xs_2xs_4s_1\underline{s_1s_2s_3s_4}$$

$$x\underline{s_1s_2}s_1xxxxx\underline{s_1s_4}xxxxs_1xs_4xs_2xs_3xxs_4xxxxs_2s_4xxxxs_3xxxxxxxxxxx\underline{s_2s_3}xx\ldots.$$

With only the above information a cryptanalysts has to construct a one time-pad $2^6 \times 4 = 256$ possibilities of combination. We can safely assume that for a large symbol source, the number of possible combinations will be larger than 2^{128}.

Hence, the one-time pad attack against the Babtista cryptosystem enhanced by the IFS matrix will be infeasible.

9.7 Experimental Result and Analysis

To judge the encryption quality of the proposed algorithm, we used two measuring techniques. These techniques are the Maximum Deviation measures, the Correlation Coefficient measures and the Strict Avalanche Criterion [8, 9].

9.7.1 The Maximum Deviation Factor

Let the plaintext encrypted by the Baptista technique be denoted by C_1 and the second encryption stage upon C_1 by the IFS matrix will generate a ciphertext denoted by C_2.

This measurement technique measures the quality of encryption based on the value between the ciphertext, C_1 and ciphertext, C_2. The more value the ciphertext, C_2 is deviated from the ciphertext, C_1 the better is the encryption algorithm. The steps involved in the measurement process are follows:

- Count the number of characters (in numerical representation) from 1 to j (based on ciphertext, C_2).
- Based on results, construct a graph (in the form of curve or histogram) for both the ciphertext, C_1 and the ciphertext, C_2.
- Construct another graph which will show the absolute difference between the ciphertext, C_1 and the ciphertext, C_2.
- Compute the sum of deviations, D between the plaintext graph and the ciphertext graph.

  This can be done by the trapezoidal rule:

$$D = \frac{h_1 + h_j}{2} + \sum_{i=2}^{j} h_i$$

where h_i is the amplitude of the absolute difference curve at value i. Since D is the sum of iterations, thus, the higher the value of D, the better is the quality of the encryption.

We will now discuss our experimental results regarding the maximum deviation factor. The following table lists down the values of the logistic map beginning from iteration-0 to iteration-80. The reader can utilize the table to verify the Baptista ciphertext value list (Table 9.1).

From the table above, we can obtain the ciphertext value list for S_1, S_2, S_3, S_4

$$S_1 = (5, 3, 1, 1, 1, 3, 11, 1, 1, 1, 1, 5, \dots)$$

Table 9.1 Logistic map iteration values for $b = 4$ and $x_0 = 0.232323$

k		k		k		k	
0	0.232323						
1	0.713396	21	0.309713	41	0.237508	61	0.537789
2	0.817848	22	0.855163	42	0.724392	62	0.994288
3	0.595890	23	0.495436	43	0.798593	63	0.022718
4	0.963221	24	0.999917	44	0.643369	64	0.088807
5	0.141706	25	0.000333	45	0.917781	65	0.323681
6	0.486501	26	0.001333	46	0.301836	66	0.875647
7	0.999271	27	0.005323	47	0.842924	67	0.435558
8	0.002913	28	0.021179	48	0.529612	68	0.983389
9	0.011619	29	0.082923	49	0.996492	69	0.065340
10	0.045936	30	0.304186	50	0.013981	70	0.244283
11	0.175304	31	0.846628	51	0.055142	71	0.738435
12	0.578291	32	0.519397	52	0.208406	72	0.772594
13	0.975482	33	0.998495	53	0.659891	73	0.702769
14	0.095667	34	0.006011	54	0.897739	74	0.835538
15	0.346058	35	0.023899	55	0.367213	75	0.549656
16	0.905208	36	0.093312	56	0.929471	76	0.990137
17	0.343227	37	0.338419	57	0.262219	77	0.039062
18	0.901689	38	0.895567	58	0.773841	78	0.150146
19	0.354585	39	0.374107	59	0.700043	79	0.510410
20	0.915418	40	0.936604	60	0.839930	80	0.999567

$$S_2 = (6, 9, 2, 2, 2, 2, 7, 9, 7, 9, 2, 8, \ldots)$$

$$S_3 = (1, 2, 9, 20, 10, 2, 4, 5, 6, 2, 10, 2, \ldots)$$

$$S_4 = (2, 2, 3, 6, 3, 2, 2, 2, 2, 7, 2, 5, \ldots)$$

A 4×4 IFS matrix key will be used. The IFS matrix is given by:

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix},$$

The encryption procedure will then be as follows:

- Start by requesting the ciphertext for S_1 we will get $S_1^* = (10, 5, 9, 7, 16, 1, 15, 13, 8, 1, 3, 7, \ldots)$.
- For S_2 we have $S_2^* = (19, 6, 17, 10, 20, 2, 11, 18, 26, 7, 18, 17, \ldots)$.
- For S_3 we have $S_3^* = (32, 1, 12, 30, 21, 10, 16, 19, 20, 6, 18, 18, \ldots)$.
- For S_4 we have $S_4^* = (13, 2, 7, 11, 9, 3, 7, 7, 16, 2, 11, 9, \ldots)$.

Based on the values after encryption by the IFS matrix key we illustrate the difference with the following graphs. The graphs illustrate the difference between

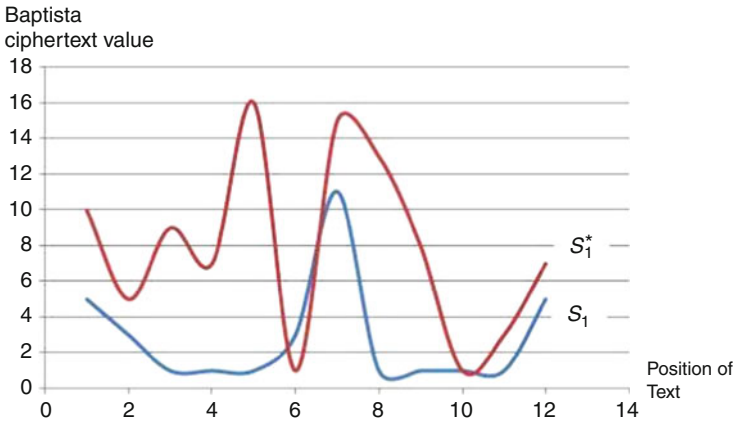

Fig. 9.1 Difference between ciphertext, S_1 and ciphertext, S_1^* with regards to position of text

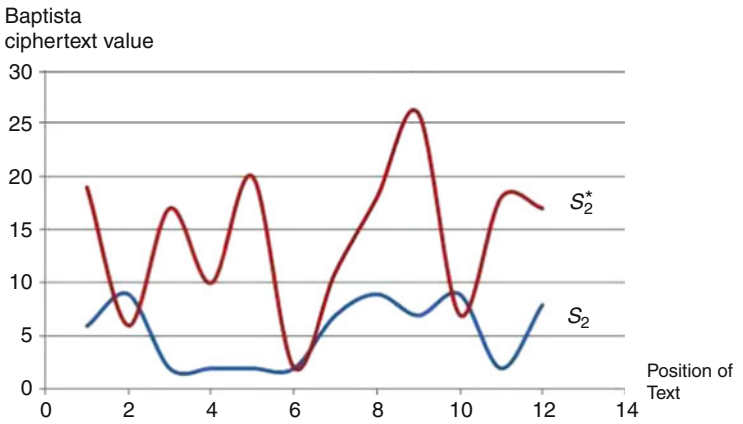

Fig. 9.2 Difference between ciphertext, S_2 and ciphertext, S_2^* with regards to position of text

the number representing the ciphertext before procedure and the number representing the ciphertext after procedure (Figs. 9.1–9.4).

The following table summarizes the maximum deviation factor for the above experiment.

By referring Table 9.2 we can observe that this algorithm produces large maximum deviation factor values.

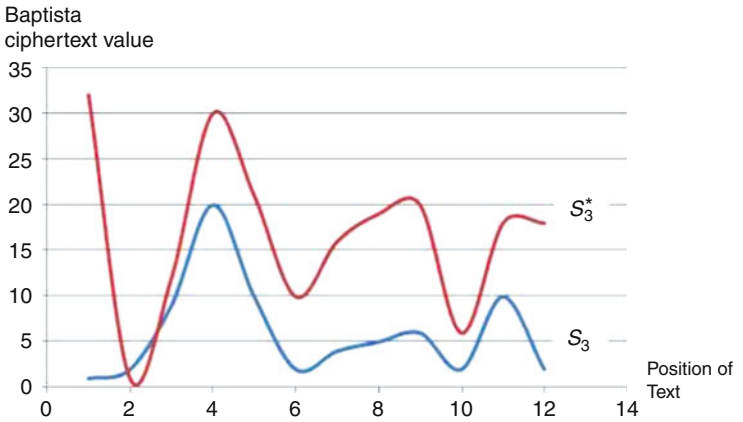

Fig. 9.3 Difference between ciphertext, S_3 and ciphertext, S_3^* with regards to position of text

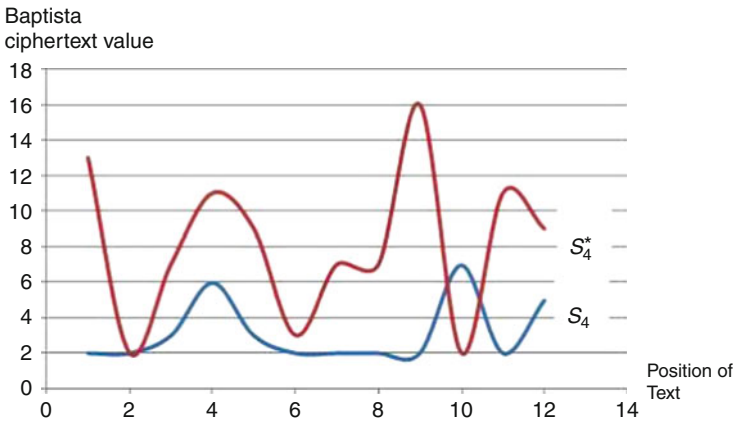

Fig. 9.4 Difference between ciphertext, S_4 and ciphertext, S_4^* with regards to position of text

Table 9.2 Measuring encryption quality via maximum deviation factor

Cipher	MDF
S_1	59.5
S_2	104
S_3	124.5
S_4	46.5

9.7.2 The Correlation Coefficient Factor

This measurement technique measures the quality of encryption based on the relationship between two variable, which are plaintext and ciphertext in this case. If the correlation coefficient is one, it means that the plaintext and ciphertext are highly dependent and the quality of encryption is not good. If the correlation coefficient is zero, it means that the ciphertext and the plaintext are not correlated that means these two variables are independent between each other. Thus, the smaller is the values of the correlation coefficient it will give us the better is the quality of encryption. The correlation coefficient (CC) can be computed by using the following equation:

$$CC = \frac{\sum_{i=1}^{N}(X_i - E(X))(Y_i - E(Y))}{\sqrt{\sum_{i=1}^{N}(X_i - E(X))^2}\sqrt{\sum_{i=1}^{N}(Y_i - E(Y))^2}}$$

The following table summarizes the correlation coefficient factor for the above experiment (Table 9.3).

The values produced are near to zero. It can be concluded that there is very little or no correlation between C_1 and C_2.

Note: A negative value of correlation coefficient means that C_1 and C_2 are negatively correlated.

9.7.3 Strict Avalanche Criterion (SAC)

The methods to test the SAC of S-boxes as introduced by Webster in 1986 and Vergili in 2001 are utilized to test the SAC of the modified Baptista cryptosystem in this article [7, 8]. Within the framework of understanding symmetric encryption, an S-box will accept an input (plaintext) and produce an output (ciphertext). Instead of accepting an input that represents a plaintext, the Baptista cryptosystem in this article accepts an input which represents the plaintext. As a result, the output produced by the Baptista cryptosystem in this article is the ciphertext. Figure 9.5 below illustrates this information.

Table 9.3 Measuring encryption quality via correlation coefficient factor

Cipher	CCF
S_1	0.33608
S_2	− 0.01759
S_3	0.45311
S_4	− 0.18728

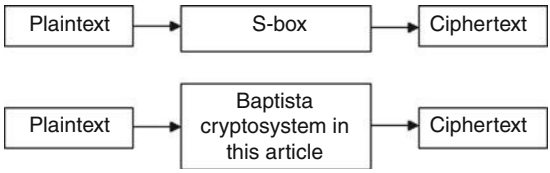

Fig. 9.5 Flowchart comparison between Baptista and S-Box

The Baptista cryptosystem in this article satisfies SAC if for all $i \in (1, 2, \ldots, k)$, $j \in (1, 2, \ldots, m)$, flipping input bit i changes the output bit j with the probability of exactly one half.

We have seen that by choosing the suitable value on initial value, x_0 the SAC value gets nearer to the idealistic value 0.5. For the benefit of the reader we re-produce definitions that are necessary in determining the SAC of a function.

Definition 4. *e^i is the unit vector with the bit i equal to* 1 *and all other bits equal to* 0.

Definition 5. *A^{e^i} is the output difference string, called the avalanche vector, when only the i'th bit in the input string is changed. It is defined as:*

$$A^{e^i} = f(x) \oplus f(x \oplus e^i) = \left[a_1^{e^i} a_2^{e^i} \ldots a_m^{e^i}\right]$$

where $a_{j^e}^i \in \{0, 1\}$.

The Baptista cryptosystem in this article satisfies SAC if for all $i \in (1, 2, \ldots, n)$ and $j \in (1, 2, \ldots, m)$:

$$\frac{1}{Q} \sum_{j=1}^{m} (a_j^{e^i}) = \frac{1}{2}$$

where

$$W(a_j^{e^i}) = \sum_{allx \in \{0,1\}^n} a_j^{e^i}$$

Take note that Q is number of terms in the summation:

$$W(a_j^{e^i}) = \sum_{allx \in \{0,1\}^n} a_j^{e^i}$$

Finally, we can compute the SAC of Baptista cryptosystem in this article by determining whether

$$k_{SAC}(i,j) = \frac{1}{Q}\sum_{j=1}^{m} W(a_j^{e^i}) = \frac{1}{2}$$

The variable $k_{SAC}(i, j)$ can take values in the range of [0, 1]. Experiments were conducted on the Baptista cryptosystem in this article with 100 samples of random initial values x_0 (see Appendix). In Table A.1 (see Appendix), it could be observed that the SAC values of each x_0 are situated within the interval [0.46, 0.54]. Thus, it is recommended that the Baptista cryptosystem in this article utilizes the value of x_0 that produces values closer to the ideal value of 0.5.

The mechanism to compute SAC for our cryptosystem is as follows:

Input:

- Set a common plaintext for each initial value.
- Set of initial values $x_0 \in \{x_{0,\,1}, x_{0,\,2},\ldots,x_{0,100}\}$ where each element of the set has a length of $round(\log_2 x_{0,i})$ *for* $i = 1,2,\ldots,100$.

Process:

- Compute $f(x)$ using the first element of initial value x_0 which is $x_{0,1}$.
- Compute $f(x \oplus e^i)$ using the first element of initial value x_0 which is $x_{0,1}$ after flipping 1 bit consecutively in the binary string.
- Compute A^{e^i}.
- Repeat steps 2 and 3 until all binaries of $x_{0,1}$ are flipped.
- Repeat steps 1 to 4 for $x_{0,\,i}$ for $i = 2,\ldots,100$.
- Finally, compute the SAC of the product.

From the Table A.1 (see Appendix), we conclude that the Baptista cryptosystem in this article has satisfied the Strict Avalanche Criterion.

9.8 Conclusion

This strengthen Baptista type cryptosystem is now resistant towards the one-time pad attack. It also has certain cryptographic properties that are favorable. Since it utilizes integers as its ciphertext, it is suitable to transmit digital information. Research on further enhancing the cryptosystem via iterating the IFS should be conducted in order to gauge the effectiveness of iterating the IFS within the Baptista type cryptosystem. Empirical results should be compared with other symmetric cryptosystems in order to further understand this unique cryptosystem. Furthermore, research should also be conducted with other chaotic maps and results focusing on security as well efficiency of these maps will give a better conclusion on which chaotic map is most suitable to be utilized via the Baptista concept together with the IFS.

Appendix

Table A.1 SAC values for selected initial values, x_0

Initial value x_0	SAC	Initial value x_0	SAC	Initial value x_0	SAC	Initial value x_0	SAC	Initial value x_0	SAC
0.4507140	0.51	0.5131530	0.48	0.3838200	0.49	0.3465580	0.47	0.7461240	0.53
0.0519714	0.47	0.2508540	0.54	0.8404540	0.51	0.9816280	0.52	0.7848820	0.48
0.8471980	0.48	0.2033390	0.51	0.4721980	0.51	0.6346130	0.52	0.4239200	0.54
0.1014400	0.52	0.8407290	0.53	0.9935610	0.50	0.7272030	0.51	0.1467900	0.50
0.6080020	0.50	0.9689330	0.54	0.4156490	0.48	0.6166080	0.53	0.8409120	0.51
0.2136230	0.48	0.3885500	0.47	0.9833370	0.48	0.9814760	0.54	0.3222050	0.50
0.7214970	0.53	0.4412230	0.51	0.9008790	0.46	0.6835020	0.51	0.2317500	0.48
0.6217350	0.48	0.6761470	0.54	0.4187930	0.50	0.9994200	0.50	0.8094180	0.51
0.9556580	0.49	0.3097530	0.47	0.1564330	0.48	0.8496400	0.46	0.5575560	0.47
0.4314270	0.53	0.3425900	0.49	0.2603450	0.47	0.2132260	0.49	0.1699520	0.50
0.3268130	0.48	0.0969849	0.52	0.9253850	0.52	0.9553830	0.48	0.3614810	0.49
0.9064030	0.51	0.8135680	0.53	0.7182620	0.47	0.0032044	0.53	0.4033200	0.47
0.1688230	0.46	0.6086430	0.48	0.3208310	0.46	0.2180480	0.53	0.8206790	0.47
0.4831540	0.52	0.2361450	0.46	0.0338440	0.53	0.4552310	0.50	0.2638240	0.50
0.4202880	0.47	0.2741090	0.52	0.4395140	0.53	0.7287900	0.53	0.3774720	0.53
0.9132690	0.48	0.0232239	0.51	0.7675170	0.51	0.5614320	0.49	0.6207280	0.50
0.1869200	0.46	0.8133540	0.48	0.1106870	0.49	0.7769170	0.48	0.8977360	0.52
0.6115420	0.48	0.9510190	0.53	0.7124630	0.50	0.4702760	0.48	0.2686770	0.51
0.5906370	0.50	0.6520390	0.53	0.6750180	0.48	0.4308780	0.50	0.3833010	0.52
0.4252930	0.50	0.7925720	0.47	0.9989930	0.51	0.7468870	0.51	0.3564450	0.48

References

1. Alvarez, G., Montoya, F., Romera, M., Pastor, G.: Cryptanalysis of an ergodic chaotic cipher. Phys. Lett. A **311**, 172–179 (2003)
2. Ariffin, M.R.K., Noorani, M.S.M.: Modified Baptista type chaotic cryptosystem via matrix secret key. Phys. Lett. A **327**, 427–430 (2008)
3. Ariffin, M.R.K., Noorani, M.S.M.: Mathematical treatment for constructing a countermeasure against the one time pad attack. In: Chaos Synchronization and Cryptography for Secure Communications: Applications for Encryption, pp. 463–475, IGI Global, Hershey PA, USA, 2010
4. Baptista, M.: Cryptography with chaos. Phys. Lett. A **240**, 50–54 (1998)
5. Barnsley, M.F.: Fractal Image Compression, Notices of The AMS, pp. 657–659, June 1996
6. Rhouma, R.: Comment on modified Baptista type chaotic cryptosystem via matrix secret key. Phys. Lett. A **373**, 3398–3400 (2009)
7. Vergili, I., Ycel, M.D.: Avalanche and bit independence properties for the ensembles of randomly chosen $n \times n$ S-Boxes. Turkish J. Electr. Eng. Comput. Sci. **9**(2), 137–145 (2001)
8. Webster, A., Tavares S.: On the design of S-boxes. In: Advances in Cryptology. Proceedings of the CRYPTO_85, pp. 523–534. Springer, New York (1986)
9. Ziedan, I.E., Fouad, M.M., Salem, D.H.: Application of data encryption standard to bitmap and JPEG image. In: Proceedings of the Twentieth National Radio Science Conference, pp. 1–8, Cairo, Egypt, 2003

Chapter 10
A Cryptographic Scheme Based on Chaos Synchronization and Genetic Engineering Algorithm

Santo Banerjee

10.1 Introduction

Genetic Algorithm (GA), developed by John Holland [1], belongs to a class of population based Evolutionary Algorithm (EA). Inspired by the natural biological model of evolution, EA has been stated to be well proven optimization technique based on meta-heuristic search paradigm [2, 3]. Based on a competitive model of survival of the fittest, the individuals of a population evolve to adapt which favors the selection of this individual in reproduction process known as *crossover*. In this phase, the genetic materials are exchanged to produce the offsprings which form the next gene pool. The diversity among species is maintained by another stochastic operator known as—*mutation*. The randomness inherent in nature can either induce stability or complete disorder. However, evolution maintains its stability where individuals identify other members of the population by special "tags." Evolutionary algorithm (EA), also known as Evolutionary computation (EC), is an umbrella term encompassing.

1. Evolution strategies (ES)
2. Evolutionary Programming (EP)
3. Genetic Algorithms (GA)
4. Genetic Programming (GP)

S. Banerjee (✉)
Cryptography Research Laboratory, Institute for Mathematical Sciences,
University Putra Malaysia, Malaysia

Department of Complexity and Network Dynamics, International Science Association,
Ankara, Turkey
e-mail: santoan@gmail.com

S. Banerjee et al. (eds.), *Applications of Chaos and Nonlinear Dynamics in Science and Engineering - Vol. 2*, Understanding Complex Systems,
DOI 10.1007/978-3-642-29329-0_10,

ES enables to find the optimal parameter setting involving continuous, discrete and combinatorial search spaces. The concept of EP emerged when Fogel et al. [4] were working with the mutation operator which changed the states of Finite state Automata. However, EP differs from ES in the selection scheme. The former conducts a stochastic tournament to determine solutions to be retained and the latter is more inclined towards following a deterministic selection approach. The Selection scheme in ES [5–10] are of four types namely [11].

- (μ, λ)(Known as Comma based strategy)
- $(\mu + \lambda)$(Known as Plus strategy)
- $(\mu/\rho, \lambda)$
- $(\mu/\rho + \lambda)$

In Comma based mechanism, the μ parents produce λ children using mutation where $\mu \leq \lambda$. Out of each of the λ children, the best become the parents for the next generation. However, in the Plus mechanism the best μ individuals which become parents for the next generation are selected from their old parents and λ offsprings. The last two canonical versions include ρ number of individuals out of μ parents in an additional step of crossover, rest of the mechanism being identical to the former two.

GA is a well proven powerful population oriented stochastic optimization technique which finds an approximate or near optimal solution to combinatorial problems. It evaluates a population of individuals, iteratively transforming the current population G(t) at iteration t into a new population G(t + 1) of offsprings. Each individual in the population participates in producing a new generation by applying the genetic operators of reproduction and gene mutation. The basic prerequisites for developing GA are a proper representation of the possible solution, known as chromosomes, as a string of symbols or as a bit sequence(in case of binary representation) and a proper choice of the three probabilistic genetic operators -selection, crossover and mutation which manipulates the population. Darwin's survival of the fittest individual is simulated in GA by a heuristic function– *fitness* which evaluates the individuals. Reproductive success is proportional to the fitness of such individuals and each chromosome competes for reproduction. However, only the best are chosen as parents in accordance to its fitness value and the offsprings so produced are added to the new gene pool. Thereby, crossover operation occurs where the worst chromosomes are replaced with the better ones through the mixing of the pairs of chromosomes so chosen. This mechanism of Crossover or mating can also occur without replacement of the best chromosome by a technique known as elitism. The scheme implies that the fittest chromosome G_{best} (t) in a generation $G_p(t)$ of p number of chromosomes is chosen as a parent and the rest of the $G_p(t)$ - G_{best} chromosomes are crossed over with it, meanwhile preserving the chromosome $G_{best}(t)$. In order to increase the domain of the search space, offsprings are mutated with very low probability. Such a single generation of EA is powerful enough to solve a wide variety of complex real world problems. However, better results may be obtained by multiple iterations of subpopulation. GA is efficient in searching a large solution space in varied directions. The noted features of GA include less computational effort, output with an optimal solution, scalability, modularity, adaptability, self

guidance by the decision metric—*fitness* and less chance of getting stuck with a non optimal solution (local optima) than back propagation neural network due to the genetic operator: *mutation*.

Since, this work focusses on ES and GA so a detailed treatment of these EC techniques are addressed. For rest of the techniques under EC, the readers are directed to refer [11]. We have proposed a technique of cryptography with noise induced two coupled Lorenz-Stenflo systems, used for key generator. Our work primarily focusses on extending the underlying principle of GA. There are certain complex problems in dynamical system for which the modification of the basic operators of the generic GA becomes imperative for efficient solution by Genetic Engineering (GE) scheme. This results in a Genetic Engineering Algorithm (abbreviated to GEA) or a hybrid GA based on the underlying principles of EA. The concept of GEA exploits the positive features of GA and finds a mechanism to eliminate the drawbacks of GA in order to enhance the quality of the solution. This objective is achieved by modifying the probabilistic operators of GA in order to render it partially or fully deterministic inspired from evolution theory which can be tailor-made to solve the problem. Thus, a hybrid version of GA is designed for specific goal-oriented problem such as in securely communicating the information. A GEA for cryptography has been developed by incorporating a learning mechanism with the help of a "tag" based learning scheme in the selection process. This serves as a guiding mechanism for the individuals during crossover phase. The selection scheme of the basic GA has been substituted with a proposed Partial Fitness based Roulette Wheel replacement strategy (PFRWR) [11] to suit our application of cryptography which is explained in subsequent sections. The methodology through which PFRWR is employed in the task of selection of parents, in the transmitter and receiver section, yields two flavors of GEA - Deterministic GEA and Pseudo-Deterministic GEA for cryptography. Comparison between these two versions has been successfully attempted. The proposed model has been compared in cryptography with $(\mu/\rho, \lambda)$-ES selection scheme which confirms the effectiveness and merit of our scheme. For this purpose the cryptosystem with $(\mu/\rho, \lambda)$ selection has also been developed.

The article is organized as follows: Sect. 10.1.1 under Introduction gives a brief overview of the interdisciplinary work between chaos and evolutionary theory together with the motivation for our scheme of cryptography. Section 10.1.2 under Introduction presents the GEA concept. Section 10.2 introduces the Key generating system which is based on synchronization of two noise driven chaotic systems. In Sect. 10.3 we have discussed our new method of encryption which is based on synchronized chaotic systems and genetic engineering. Section 10.4 reports the conclusion of our results.

10.1.1 Background and Motivation

Chaos theory and Evolution theory share rich features which can form powerful tools for various applications when exploited. It is well known that chaotic systems

are dynamical in behavior describing the complex non linear system which after several iterations reveals a pattern of a strange attractor whose behavior can be monitored. The dynamic behavior in GA is induced together by crossover and mutation operators. The recombination operator cuts and joins pieces of information from different individuals to yield a better overall solution. These operations affect the overall fitness of the gene pool. This working principle is synonymous to an emergent dynamical system wherein the influence of the GA operators can either stabilize or destabilize the search space.

Furthermore, EA and chaotic systems can be simulated. Simulation of random and dynamic behavior inherent in organic evolution can be realized in EA by using a pseudo-random number generator combined with selection based on a decision metric *fitness*. Also a simulated version of reproduction can be realized by *crossover* and the *mutation* operator. When the random changes of individuals and fitness based selection are combined then the computerized system will be able to solve the problem faster than a random search. For example, the changing behavior of a population over generations can be visualized through computer graphics similar to the Mandelbrot set. In light of these facts, this section uncovers a comprehensive overview of the interdisciplinary work between chaos and evolutionary theory.

EA are meta heuristic search techniques and not a directed search technique because biological evolution itself is not purposeful or intuitive. GA is probabilistic because initial population is generated randomly, selection of parents for mating occur randomly, fitness determination can also be probabilistic in case of Monte Carlo method and crossover points and mutation rates are determined randomly.

Recent trend shows a subtle deviation from the traditional schemes of cryptography towards techniques which are self learning with reduced computational overhead. There has been considerable developments in the area of cryptology based on EC techniques especially the stochastic properties of GA which have been successfully exploited in cryptography [12] and cryptanalysis. Due to the fundamental characteristics of sensitivity to initial conditions, ergodicity [13], deterministic yet unpredictable nature and mixing, Chaos theory finds its applicability in secure communication [14, 15]. Chaos based systems are defined on real numbers only whereas cryptographic systems operate on finite number of integers. Chaos theory models and solves real world problems analytically. EA belongs to an emerging derivative of Artificial Intelligence known as Computational Intelligence (CI) [16]. CI borrows examples from real world problems to examine their working principle so as to yield solutions to complex NP hard problems. Chaos modeling of GA can be another example of the potential use of chaos as a tool to analyze GA. The areas of artificial neural networks, fuzzy logic and GA can be employed together with chaotic systems with or without the help of fuzzy logic. This concept is called a Hybrid System. These hybrid versions of GA may be combined with other Computational intelligence offshoots to effectively solve complex goal oriented problems with improved performance. There have been noteworthy attempts to combine dynamical features of evolution with chaos theory [17]. Recently, there is a novel trend towards exploring hybrid systems and hybrid versions of GA whose

genetic operators may be modified to suit the problem [18, 19] giving promising result. The next section lays the foundation for adopting the modified GA by GE technique.

10.1.2 The Genetic Engineering Scheme for Cryptography

The objective of the article is to introduce and discuss the concept of a GEA together with deterministic chaotic system serving as the key generator. The basic GA imitates natural evolution. The operators of GA are probabilistic which is advantageous for exploration of the solution space in varied directions. However, GA is marred by excessive long run-time and it often risks in getting trapped with a suboptimal solution affecting the efficacy of real time complex problems. This demands a substitution of the basic operators of GA to yield an efficient solution to the problem by genetic engineering (GE) scheme. When the stochastic operators are perturbed to present a genetic representation specifically designed to address certain class of problems, we denote such tweaking as genetic engineering of the operators. This procedure results in a Genetic Engineering Algorithm (GEA) which is an extension of the basic GA. The term is derived from the current branch of modern biology-genetic engineering. The genetic engineer identifies a superior genetic material and uses this to explicitly classify the population into a genetically altered superior group and other sub-groups. Thus, the modification of GA adheres to the Darwinian evolution model but at the same time, it is doped with genetic engineering technique. Such enhancements aid in exploiting the positive features of GA and to devise a mechanism to eliminate the drawbacks of GA, in order to enhance the quality of the solution. GEA is inspired from GA and has been applied effectively for solving complex goal oriented problems.

An exploration oriented optimization solution can be derived through a proper *Selection* mechanism. The stochastic selection operator of GA can be usefully tapped to perform a blind search of the solution space in varied directions to reach an optima. There are three most commonly used selection mechanisms of GA. The roulette wheel selection method always selects a chromosome which dominates over the rest because the individual with the highest fitness score gets a priority in being selected as a parent. Again in Tournament selection, the best individual from a "tour" is selected to mate with the next fitter individual. In the Rank based scheme parents are chosen based on the rank index only. In our GE scheme, the selection scheme is not purely based on the greedy method where always the fittest individual is chosen as a parent which is the case of GA selection mechanism especially in Rowlette Wheel (Fig. 10.1).

But randomness can sometimes destabilize the system which is an undesired characteristic for optimization problems. Moreover, there is no guarantee that it will yield an improved offspring. Also, convergence speed can be slow which is undesirable for many complex optimization problems. GA is rendered dynamical by its stochastic operators which brings drastic changes and can yield a near optimal

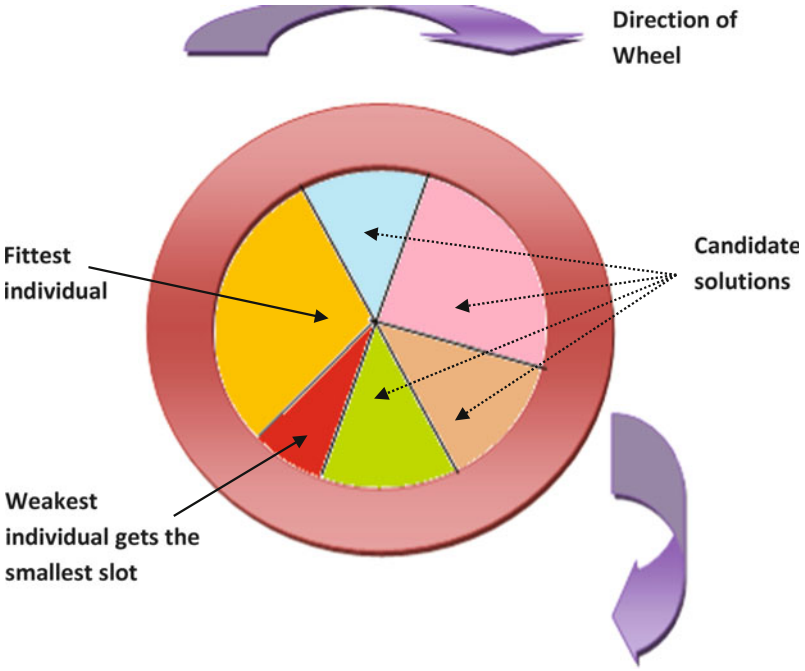

Fig. 10.1 (Color online) Schematic diagram of a Roulette Wheel

or no solution at all. We opt for a tradeoff by employing the pros and cons of Rowlette Wheel. The controlled randomness has been achieved by injecting deterministic features into the selection scheme of generic GA by incorporating a "tag" based learning mechanism into the chromosome structure by GE technique. This leads to a partial fitness based selection scheme emerging with replacement of the fittest individual (known as PFRWR). The altered version adheres to the fact that evolution maintains its stability where individuals identify other members of the population by a complex gene residing in the chromosome. This tag serves as a learning mechanism and guides the individuals of the population to perform their genetic operations by choosing this superior chromosome as a candidate for crossover operation thus eliminating premature convergence problem of GA. The proposed hybrid version of GA will render the GEA partially or fully deterministic giving way to a novel, reliable and robust GEA scheme which evolves itself to produce the desired result. The way selection operation is carried leads to the following two novel ways to achieve an efficient cryptography. For cryptographic purposes, the best chromosome is selected from the available candidate solutions as the cipher text (CT) which is also recovered fully. The first method—Selection1 renders the GEA deterministic whereas, the second method—Selection2 makes the GEA pseudo deterministic. The former is a self learning method where the "tag"

remains with the successive offsprings of the fittest parent initially chosen to instantiate crossover. This also happens in Stochastic hill climbing which is a variant of hill climbing approach. When the candidate solution having highest fitness is chosen during both encryption and decryption process, it implies that all the solutions are compared first and then a decision is made to move to the one which is closest to the goal or to move to another sample point. This is akin to a spin of the Roulette wheel where all the individuals in the population are checked for the contesting individual to become the first parent. In Pseudo-Deterministic GEA, the tag is released after the first crossover in both the transmitter and receiver's end and for the rest of the crossovers, the process wanders randomly through the discrete search space which contains both the good and bad solutions. The end result is a successful termination with the global optimal solution. The evolutionary process of the proposed GEA makes the individuals of sub-generations distributed ergodically in the defined space converging towards the goal of selecting a "good" quality Cipher Text.

The key generating system is a noise induced coupled system which on synchronization attains a chaotic identical state. Both the system can act as a Transceiver. Thus the two communicating parties form a secure communication channel [20]. The chaotic sequences are used only once with an XOR mode for cryptography and such a stream of data is truly random forming a one-time pad which is theoretically unbreakable.

10.2 Noise Induced Synchronization Between Two Chaotic Systems

10.2.1 The System and Its Chaotic Properties

We consider the Lorenz Stenflo system [21, 22], driven by two additive white noise [23, 24]

$$\begin{aligned} \dot{x} &= a(y - x) + cw \\ \dot{y} &= x(r - z) - y + D\xi(t) \\ \dot{z} &= xy - bz + D\xi(t) \\ \dot{w} &= -x - aw \end{aligned} \tag{10.1}$$

Where x, y, z, w are the state variables and a, b, c, r are parameters of the above system. The deterministic model exhibits chaos for a = 2.0,b = 0.7, c = 1.5 and r = 26.0. Here $\xi(t)$ is a Gaussian white noise with the intensity D. Figure 10.2 represents the simulation analysis of the white noise $\xi(t)$. The upper panel

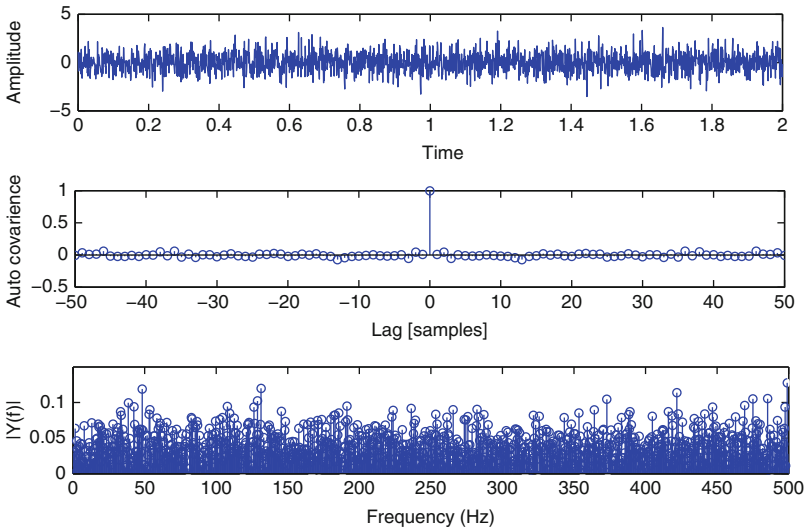

Fig. 10.2 (*Upper panel*) The amplitude of the generated noise, plotted against time, (*middle*) the auto-covariance of the white noise, plotted against lag L, (*lower panel*) the modulus of noise spectrum, plotted against frequency f

represents the amplitude of the noise with zero mean and unit variance. The middle is the covariance of the generated white noise. The lower diagram is the frequency domain of the noise. Figure 10.3 represents the colormap of the bifurcation diagram of the system (1) with respect to the parameter r.

10.2.2 Synchronization in Presence of Noise

We now consider the synchronization between system (1) and the following identical system driven by the common additive noise $\xi(t)$.

$$\begin{aligned} \dot{x_1} &= a(y_1 - x_1) \\ \dot{y_1} &= x_1(r - z_1) - y_1 + D\xi(t) \\ \dot{z_1} &= x_1 y_1 - bz_1 + D\xi(t) \\ \dot{w_1} &= -x_1 - aw_1 \end{aligned} \tag{10.2}$$

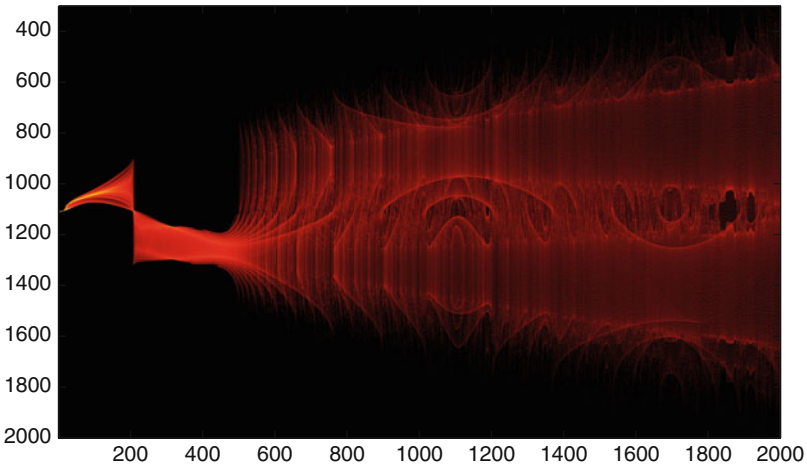

Fig. 10.3 Bifurcation diagram of the system (1) in presence of noise

The purpose of adding noise is to eliminate coupling terms, since they contain info about the driving signals and the state variables as some of them are transmitted. Synchronization in presence of common noise does not transmit any information about the state variables and system parameters. The external noise is known to both the transmitter and receiver. Let the error vectors are defined as

$$e_1 = x_1 - x, e_2 = y_1 - y, e_3 = z_1 - z, e_4 = w_1 - w \tag{10.3}$$

Figure 10.4 represents the time variation of the error vectors for D = 5. It can be observed that all the error vector components tends to zero after $t > 20$.

We will apply the two synchronized chaotic systems for digital cryptography [25–27]. At the synchronized state, the sender uses the system (1) and the receiver use the system (2) to construct their secret keys. The steps of our proposed scheme is discussed in the next section.

10.3 Proposed Cryptography

This section explains the GEA scheme for the purpose of cryptography inspired from GA. The key sets used for encoding and decoding are derived from the previous section explained above. In our scheme of GEA for the encryption, initialization of the population by breeding the different encoded versions of the cipher text to generate the initial population is also an essential operator apart from the rest of the GA operators. Subsequent sections explain the proposed GEA for cryptography.

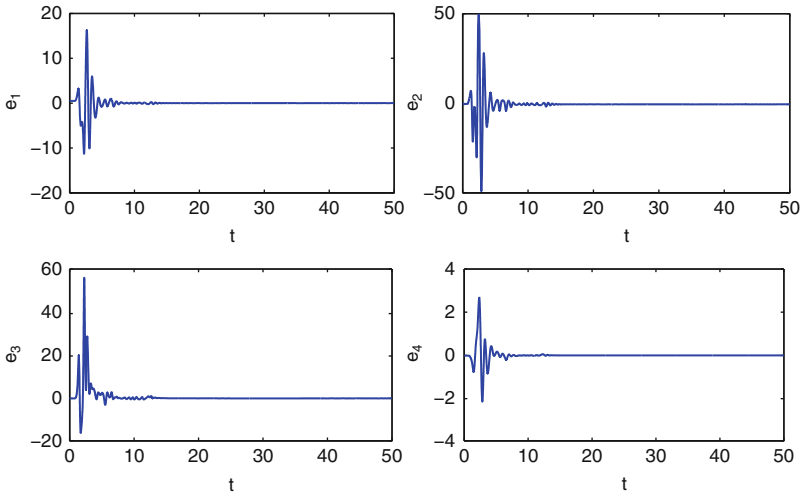

Fig. 10.4 Time variation of the error systems showing synchronization

10.3.1 Design of GEA

A predetermined number of individuals n are generated from the CT using a generator function G_n to produce valid solutions based on CT. The initial population G_n (0) of encrypted chromosomes is populated using integer values obtained from the chaotic sequences generated by the system described in the previous section. The individuals are then evaluated with the maximizing objective function—*fitness*. The selection mechanism for the proposed GEA using the scheme known as Partial Fitness Rowlette Wheel Replacement strategy (PFRWR) together with a learning mechanism devised with a concept of "tag" has been developed and illustrated in. The way selection differs using fitness as a partial heuristic function for decision yields two versions of GEA algorithm viz Deterministic GEA (Fig. 10.5) and Pseudo-Deterministic GEA (Fig. 10.6) which has been successfully applied in cryptography as seen from results in the subsequent sections.

10.3.2 Methodology and Application in Secure Communication

This chapter employs the features of natural biological evolution and its transformation operators viz selection, crossover and mutation by adhering to its inherent meaning. But the way they subsequently mix their information is modified to increase the reliability and decrease the uncertainty of recovery of the message.

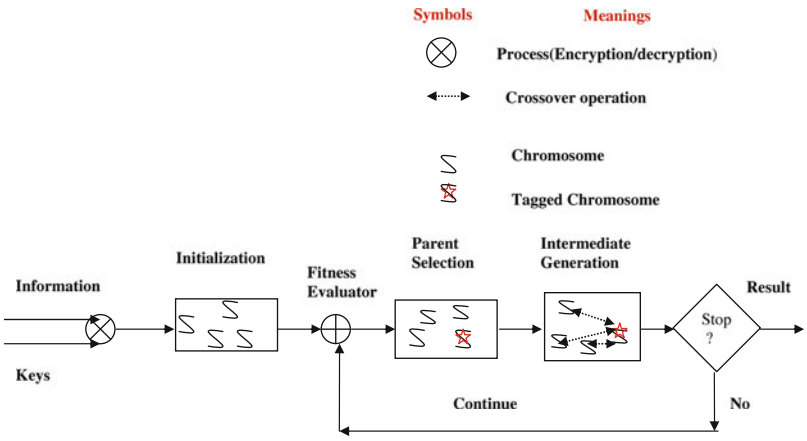

Fig. 10.5 (Color online) Schematic diagram of deterministic Genetic Engineering algorithm with PFRWR selection

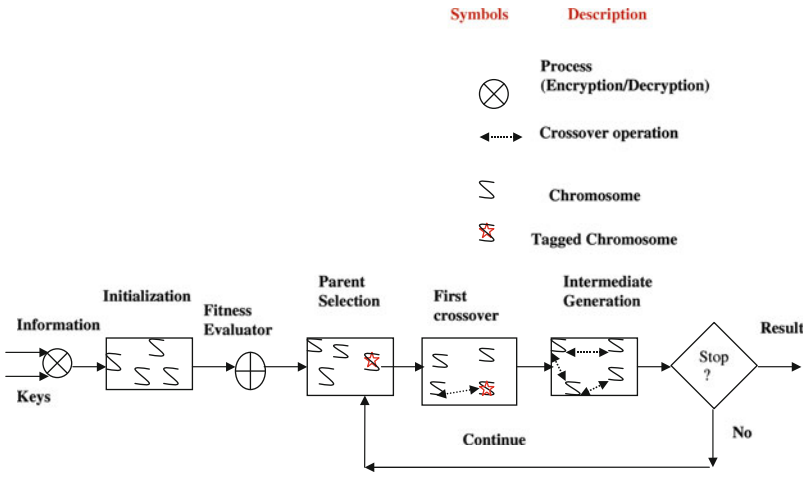

Fig. 10.6 (Color online) Schematic diagram of pseudo-deterministic Genetic Engineering algorithm with PFRWR selection

The discrete solution space is represented by encoded and decoded chromosomes. The objective is to choose a strongly encrypted string of better quality relative to the available candidate solutions. This choice is guided by the fitness measure and the selection schemes discussed in the previous subsection. The selection process has

been closely monitored by implementing both the proposed PFRWR selection schemes for the population. Selection is governed by a maximization objective function *fitness*. Initially, each letter of the plaintext M is encrypted. M is XORed with n × population size(denoted as *size*) number of keys obtained from synchronization of chaotic systems. The key sets keys = $\{k_1, k_2, \ldots k_n\}$ used are of the same length n as that of M to generate the CT. Encryption technique adopted is Vernam cipher. In our case, the heuristic function is defined as sum of how many steps or characters away each encrypted gene is from the message M. The encoding and decoding resulting in change of an element of a chromosome is synonymous to mutation in evolution. In evolution theory, mutation is defined as a change in the DNA sequence of a gene. The process of encryption can be considered as being subjected to several point mutations wherein each element of M is changed during encoding and decryption to inversion mutation scheme where every element of the CT undergoes an inverse of the encryption scheme so adopted. For bit based representation, point mutation will result in flipping the selected gene. Since it is a problem specific GEA, a certain amount of deterministic factor is injected and we have assumed that all successive crossovers till the termination of the scheme occurs with a fixed crossover point C_p and C_{p1} and C_{p2} generated randomly for one point and two point crossover respectively.

The algorithm for Deterministic and Pseudo-Deterministic GEA for Cryptography with the modified selection mechanism is presented below

1. Transmitter Section

1.1 Initial population :

Initialization:-Decide a predefined number of generations t and the population size *size*

Create initial population G_p (t) of encrypted chromosomes, where p = 2, 3,. . ., size and t = 0 of fixed length n from message $M = \{m_0, m_1,\ldots m_{n-1}\}$ using the series of keys K = $\{k_1, k_2, \ldots, size\}$ by performing an XOR operation on M. The keys used are generated from the system described by D. Ghosh et al. [27]

Set counter for keys k = 0;
For i = 1, 2, . . ., size and
For j = 1, 2, . . ., n
$Chromosome_{i,j} = M_j \bigoplus keys_{k++}$

1.2 Evaluate fitness
Set fitness for all individuals $fitness_i = 0$
For each i = 1, 2, . . ., size and
For each j = 1, 2, . . ., n
Calculate $fitness_i = fitness_i + \text{abs}(M_j - \text{Chromosome}_{i,j})$
Find best fitted chromosome $Chromosome_{best}$

1.3 Parent selection and Crossover:-
Select parents based on either
(a) **Selection1**-The Deterministic GEA
(b) **Selection2**-The Pseudo-Deterministic GEA

Select type of crossover scheme. The system has been tested effectively with one-point and two-point crossover scheme and results have been shown in subsequent section

For i = 1, 2, …, size perform crossover between $Chromosome_{best}$ and $Chromosome_i$ It is to be noted that no crossover occurs with itself.

1.4 Termination of encoding:- Terminate the encryption algorithm after a fixed number of iterations depending upon the parameters/user choice.

The ciphertext $Chromosome_{best}$ and the crossover point are transmitted which are also encoded.

2. Recovery of plaintext

2.1 Fitness of the CT ($Chromosome_{best}$) is evaluated into $fitness_{cipher}$.

2.2 Initialization :-Breed the same number of individuals by performing inverse mutation which is decoding the CT with the key set used in encoding by performing XOR operation.

2.3 Evaluate fitness:-Evaluate fitness of decoded chromosomes as discussed in step 1.2.

2.4 Parent selection and Crossover:-This step is the same as 1.3.

2.5 Termination of the process:-

Tune fitness of decoded chromosomes to achieve fitness of $fitness_{cipher}$

For each i = 1, 2, …, size Terminate when $fitness_{cipher} = = fitness_i$ else continue.

The proposed scheme has been tested with encryption with the (μ/ρ, λ) Selection scheme. The test system has been developed explained in a nutshell as under.

All the steps mentioned in Transmitter and Receiver end are the same except for a change in Step 1.3 and 2.3. For cryptography with (μ/ρ, λ) Selection and crossover is as follows

(a) Choose the chromosome which has the highest fitness value as one of the parents and denote it as $Chromosome_{best}$. The other parent is randomly selected from the generation of μ individuals.

(b) Perform crossover replacing the ρ parents with λ offsprings. Here $\rho = 2$ and $\lambda = 2$.

(c) Evaluate fitness of the λ offsprings. Select the offspring with higher fitness as the parent for remaining crossovers in the current generation. This selected offspring is now $Chromosome_{best}$.

(d) Return the other offspring into the next generation pool.

During decryption, the fitness measure is used to direct the individuals to adjust their fitness value to match with that of the calculated fitness $fitness_{cipher}$. Hence, this heuristic operator gives an estimate of how far we are from the goal of successfully recovery of PT. This is an essential property of evolution where the individuals are self learning and adapt themselves to solve a problem in hand. The algorithm is simulated with one point and two point recombination scheme and a comparison of successful recovery with the (μ/ρ, λ) apart from other observations have been discussed in the following subsections.

A flowchart for the proposed cryptography is illustrated in Fig. 10.7

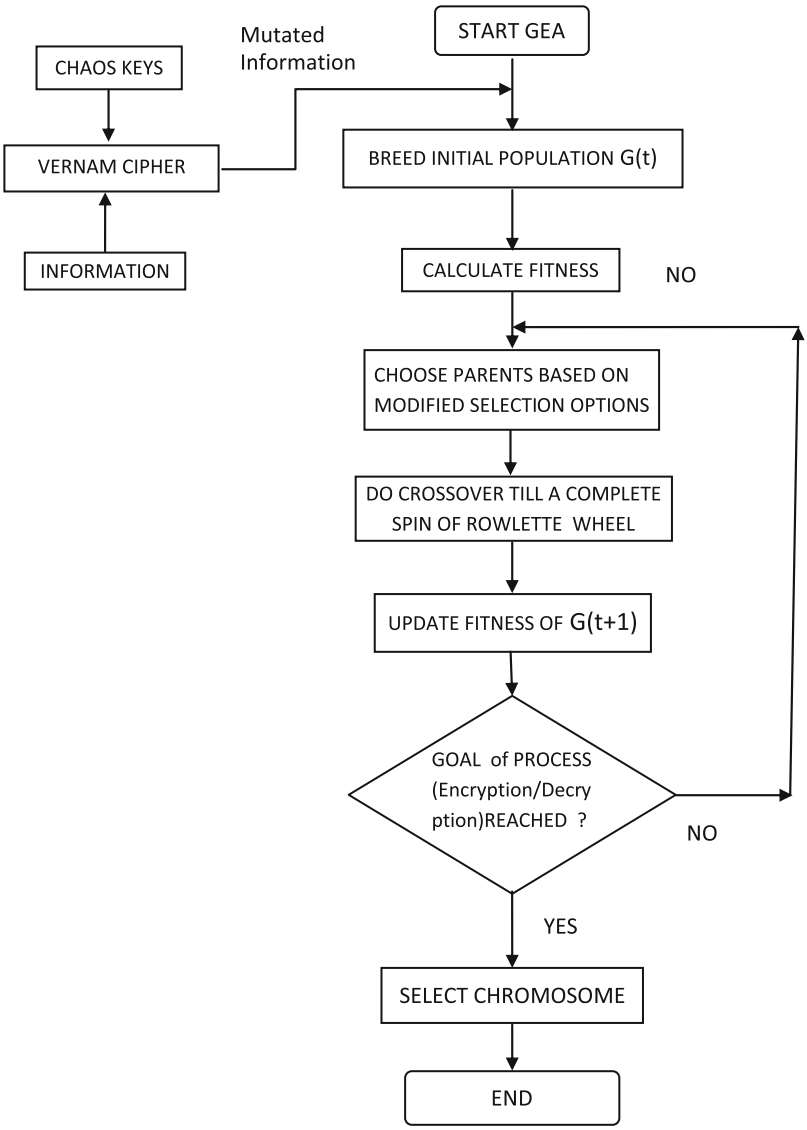

Fig. 10.7 (Color online) The flow chart of the proposed cryptography

10.3.3 Result

The execution of the above steps is illustrated with a small example in this section. Cryptography is demonstrated with Deterministic and Pseudo-Deterministic GEA with one point crossover recombination method for a small population of three

chromosomes which are subjected to three generations at the transmitter section. Suppose the sender wants to send the message

Hello123

to the receiver. A random crossover point C_p is generated.

The key sets are generated from the synchronized Lorenz-Stenflo systems (1) and (2). The sender use the x-variable of the system (1) and the receiver uses the corresponding variable of the system (2). They both pre determine to choose the corresponding numerical values at t = 5, 6, 7 respectively. To convert the keys into integer, the sender and receiver both pre determine to process the data set by following simple rule of [1000 x] (Mod 38) and [1000 x_1] (Mod 38) respectively [26, 27]. Therefore the first key $k_1 = 13$ is the value of x at t = 5 with the transformation [1000 x] Mod(38) and so on.

Following key sets are chosen $k_1 = 13$, $k_2 = 18$, $k_3 = 23$, $k_4 = 23$, $k_5 = 22$, $k_6 = 27$, $k_7 = 30$, $k_8 = 30$, $k_9 = 21$, $k_{10} = 29$, $k_{11} = 32$, $k_{12} = 35$, $k_{13} = 36$, $k_{14} = 17$, $k_{15} = 19$, $k_{16} = 25$, $k_{17} = 5$, $k_{18} = 33$, $k_{19} = 27$, $k_{20} = 18$, $k_{21} = 21$, $k_{22} = 29$, $k_{23} = 6$, $k_{24} = 7$ and one $k_{25} = 5$ for encrypting C_p.

The initial population $G_p(0)$ of encrypted chromosomes derived through Vernam encryption technique and their associated fitness is as under, which in many cases, is enough to keep things from prying eyes.

CRYPTOGRAPHY WITH Selection1		CRYPTOGRAPHY WITH Selection2	
Population	Fitness	Population	Fitness
Ew{{*y**, –	80	*Ew*{{*y**, –	80
]*xLOK*!*	180	]*xLOK*!*	180
MDw ~ *z*, 44	86	*MDw* ~ *z*, 44	86

First time crossover occurs with the best fitted parent with fitness 180 in both the cases. But subsequent crossovers shown below shows the obvious difference in both the scheme.

CRYPTOGRAPHY WITH Selection1			CRYPTOGRAPHY WITH Selection2		
Gen	Pop	Fitness	Gen	Pop	Fitness
1	*Ew*{*OK*!*	144	1	*Ew*{*OK*!*	144
.	]*xL* ~ *z*, 44	109	.	]*xL* ~ *z*, 44	109
.	*MDw*{*y**, –	93	.	*MDw*{*y**, –	93
2	*Ew*{{*y**, –	80	2	*Ew*{*OK*!*	144
.	]*xLOK*!*	180	.	]*xL*{*y**, –	116
.	*MDw* ~ *z*, 44	86	.	*MDw* ~ *z*, 44	86
3	*Ew*{*OK*!*	144	3	*Ew*{{*y**, –	80
.	]*xL* ~ *z*, 44	109	.	]*xL* ~ *z*, 44	109
.	*MDw*{*y**, –	93	.	*MDwOK*!*	157

In the receiver section for both the schemes, the PT for the Deterministic GEA is recovered immediately after one iteration. Whereas the Pseudo-Deterministic GEA takes three iterations to recover the PT. This small example highlights that the Deterministic GEA although yields a CT of lesser quality than the other, it takes

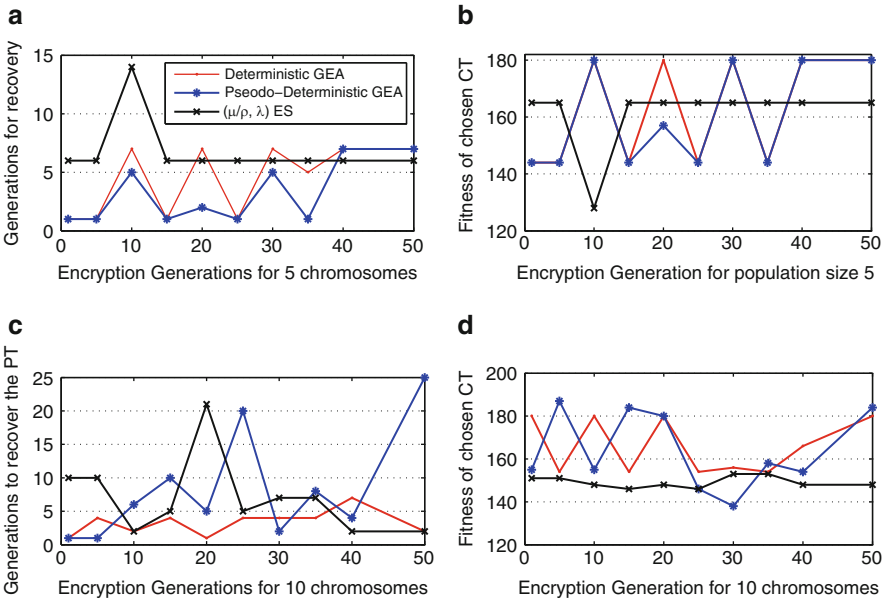

Fig. 10.8 (Color online) Comparison of deterministic Genetic Engineering algorithm, pseudo-deterministic Genetic Engineering algorithm and $(\mu/\rho, \lambda)$ in one-point crossover: (**a**) Comparison of Convergence with population size 5 (**b**) Comparison of fitness score with population size 5 (**c**) Comparison of Convergence with population size 10 (**d**) Comparison of fitness score with population size 10

lesser iterations to recover. However, this may not always hold true for all cases which has been discussed in the next section. Since the Pseudo-Deterministic selects parents randomly for subsequent crossovers, it may be possible that an individual may not get a chance to be picked for mating whereas in the cryptography with Deterministic GEA, every individual participates in mating. It has also been observed with several test cases that after two or three crossovers the same set of offsprings gets created for small sample sizes such as in this example. This happens because the domain for mixing is limited to such small number of individuals.However, it will be inappropriate to generalize this observation for all since much of the evolution depends on the fitness, population size and selection mechanism. In the following sub-section, we discuss and analyze the simulation results. The color codes viz. red, blue and black in the graphs are used to indicate the probabilistic data points for cryptography with Deterministic GEA, Pseudo-deterministic GEA and cryptography with $(\mu/\rho, \lambda)$ ES scheme respectively. The best encrypted chromosome(CT) obtained from Cryptography with Deterministic GEA is of fitness 144 whereas its counterpart yield a CT of fitness 157. This is the essence of evolution that better offsprings are

produced here. The reverse happens at the receiver's end. Although the number of crossovers and generations are not transmitted, the algorithm converges successfully faster than the $(\mu/\rho, \lambda)$ variant which can be observed from Fig. 10.8.

10.3.4 Observation

This section shows the experimental results on simulation of the proposed cryptography by varying the population size of fixed length PT subjecting each population to 50 iterations. Simulation is performed with one point and two point crossover technique. Then, a comparison is made with the cryptography developed by $(\mu/\rho, \lambda)$ ES selection scheme to demonstrate the marked performance improvement and the efficacy of the encryption technique with the proposed GEA.

1. *Convergence and Quality of cipher text* :

- Figure 10.8 shows a comparison between the proposed cryptography and cryptography designed with $(\mu/\rho, \lambda)$ ES selection.
  The probabilistic data points in Figs. 10.8a,b and 10.7c,d correspond to sample size of 5 and 10 respectively. From Fig. 10.8a it can be observed that $(\mu/\rho, \lambda)$ attains a maxima when the number of encryption generations are 10 implying that this takes more number of iterations to evolve. However, after transmitter generation 15, it takes same number of steps to decode the CT in comparison to the proposed scheme of cryptography. Moreover,$(\mu/\rho, \lambda)$does not effectively explore the search space and yields a CT having fitness 165 after generation 15 in Fig. 10.8b with a fitness score of 153 in Fig. 10.8d. This is in contrast to the proposed scheme which delivers solution having more or less the same fitness for population size 5. The coinciding fitness values demonstrate that for small population size, Deterministic and Pseudo-deterministic GEA perform equally well in exploring the search space. The fitness landscape resembles that of a hill climbing approach with several high and low peaks because after every encryption cycle the linkage and randomness influences the overall fitness of the individual. Figure 10.8c,d show that the population in $(\mu/\rho, \lambda)$ ceases to evolve emphasizing that the proposed cryptography with the PFRWR selection scheme adheres to the evolutionary strategy wherein every generation evolves itself to produce fitter offsprings. Figure 10.8b shows that Deterministic GEA has more or less low peaks throughout the generations. Pseudo-deterministic takes more iterations to terminate which is compensated for it yields a better solution at generation 5 and 50 as observed from Fig. 10.8d. This is a tradeoff. It is noteworthy, that at transmitter generation 5, Pseudo-Deterministic GEA already yields the optimal CT of value 187. Subsequent evolutions have not yielded a candidate solution which dominates the obtained solution. This throws insight over the fact that for the pseudo deterministic cryptography, for a small sample size, GEA yields the optimal solution. However, Pseudo-Deterministic GEA

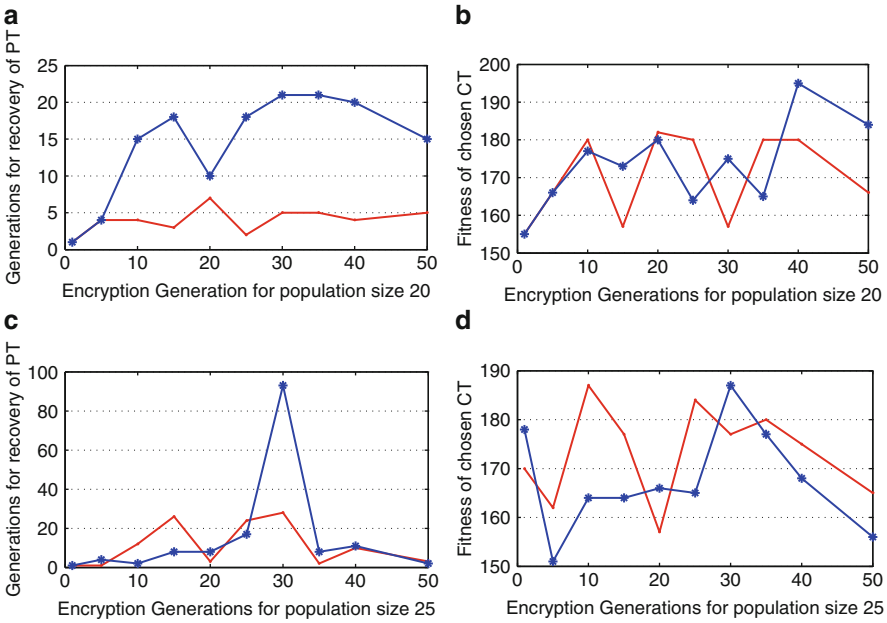

Fig. 10.9 (Color online) Comparison of deterministic Genetic Engineering algorithm and pseudo-deterministic Genetic Engineering algorithm in one-point crossover: (**a**) Comparison of Convergence with population size 20 (**b**) Comparison of fitness score with population size 20 (**c**) Comparison of Convergence with population size 25 (**d**) Comparison of fitness score with population size 25

gives a better solution than Deterministic GEA with both the proposed versions subtly escaping points of local optima.

- Figure 10.9a–d represents a comparison between cryptography with Deterministic and Pseudo-Deterministic GEA with probabilistic data points corresponding to the population size 20 and 25 respectively. Figure 10.9a shows that the Deterministic GEA has a maxima at transmitter generation 20 after which the peak is below 5 all throughout. It is noteworthy from Fig. 10.9b that Pseudo-Deterministic solution takes more iterations to recover the PT, but it also yields the best CT at generation 40. In Fig. 10.9c, it is observed that Deterministic option has overall low peaks and its counterpart attains a maxima at generation 30 in transmitter section for population size 25. Whereas, for the same number of generations at encryption side, Deterministic GEA recovers the PT after 28 iterations. Figure 10.9d reveals that Deterministic GEA has a better fitness contour. It yields an optimal solution of fitness 187 for the entire population at generation 10.

  In Fig. 10.10a we can see that both the proposed solutions for cryptography have prominent maxima and minima, out of which Pseudo-Deterministic GEA attains the highest peak. The same behavior is observed in Fig. 10.10c at encryption

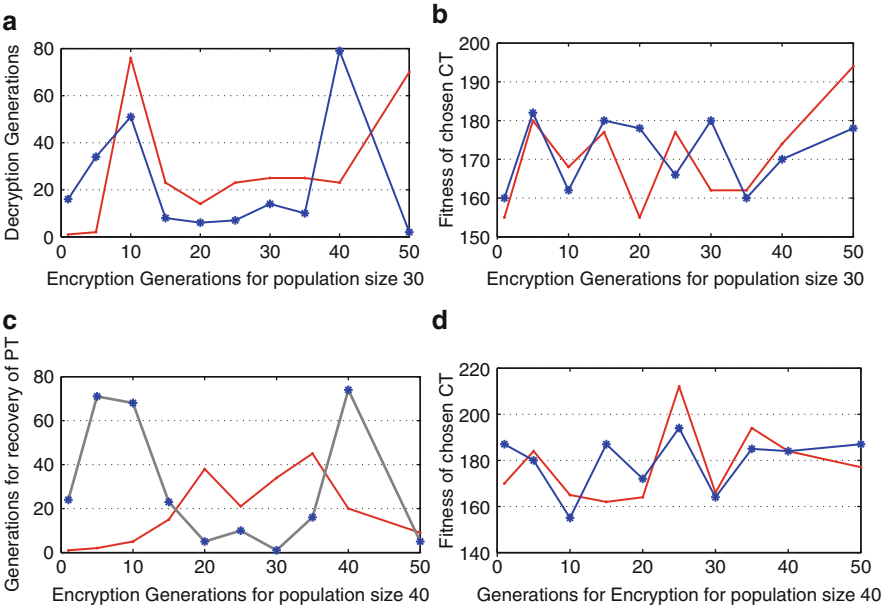

Fig. 10.10 (Color online) Comparison of deterministic Genetic Engineering algorithm and pseudo-deterministic Genetic Engineering Algorithm in one-point crossover: (**a**) Comparison of Convergence with population size 30 (**b**) Comparison of fitness score with population size 30 (**c**) Comparison of Convergence with population size 40 (**d**) Comparison of fitness score with population size 40

generation 40. Even though, in Fig. 10.10b, Pseudo-Deterministic GEA explores more search space, it does not yield a "better" candidate solution. Figure 10.10d also confirms this observation. In this case, Deterministic GEA attains a maxima at generation 25 yielding the optimal CT of highest fitness measure(212) out of all the test cases. Therefore, the overall behavior is unpredictable. This fact is an attribute of dynamical system and hence no general consensus can be reached as to when which option outperforms the other. Nevertheless, we may infer that if randomness and probabilistic behavior of GA can be controlled then a reliable CT may be obtained.

- Figure 10.10 shows the probabilistic data points of the two selection schemes of GEA in cryptography with two point recombination. In Fig. 10.11a Pseudo-Deterministic GEA attains a maxima at encryption generation 25. So it takes more iterations to terminate with the decoded CT in contrast to Deterministic GEA for sample size 5. Figure 10.11b shows that for the same number of encryption generation of 50, Pseudo-Deterministic GEA takes maximum number of iterations to recover the message. This implies that random process takes considerable time to stabilize and converge in contrast to a controlled strategy which also emphasizes the importance of randomness in achieving an

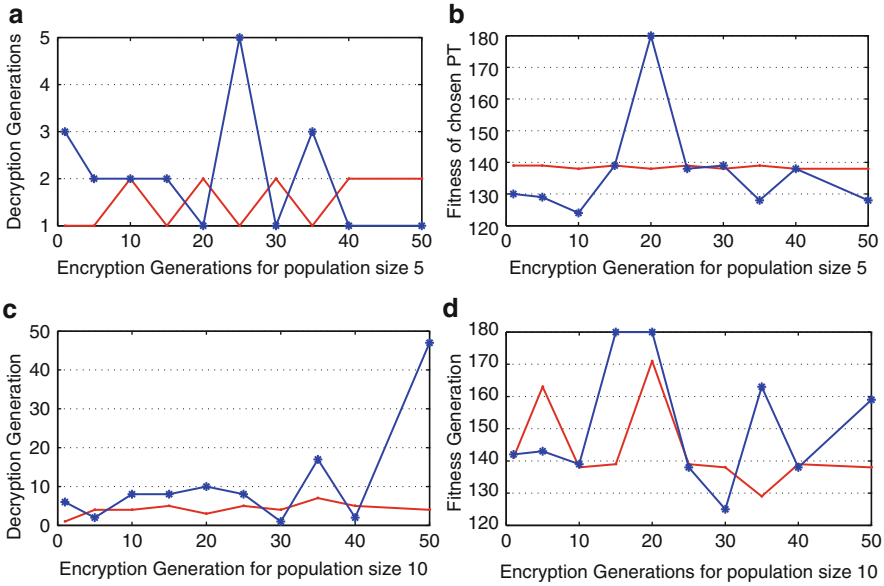

Fig. 10.11 (Color online) Comparison of deterministic Genetic Engineering algorithm and pseudo-deterministic Genetic Engineering algorithm in two-point crossover: (**a**) Comparison of Convergence with population size 5 (**b**) Comparison of fitness score with population size 5 (**c**) Comparison of Convergence with population size 10 (**d**) Comparison of fitness score with population size 10

exploration oriented search achieved through Pseudo- Deterministic strategy. In Fig. 10.11d a "plateau" like flat landscape is observed with a two-point crossover which is a major drawback in hill-climbing. These regions have points with similar heuristic value(fitness) which often leads to abrupt termination of the process. In a way the tag in this case transforms the Pseudo Deterministic solution into a simulated annealing algorithm after the first crossover in a generation. On closer examination,the proposed scheme is quite similar to Simulated Annealing but with a major difference. In Simulated Annealing, emphasis is laid on modifying the solution by one move which is similar to the function of mutation operation in GA. Whereas, in GA a solution is achieved by taking combinations of existing solutions. In general, the major drawback of premature convergence by settling with a poor offspring in GA is solved by using PFRWR selection scheme as a selection operator for the proposed cryptography. In GEA for cryptography, as observed from Fig. 10.10d, the Pseudo-Deterministic GEA escapes a plateau region by a steady gradient descent, hits a solution at generation 30 evolves at further generations and gives a candidate solution of fitness value 163 at iteration 35. This is also because elitism has been switched off. So the proposed scheme shows its merit by achieving

exploration-oriented search in avoiding local minima and exploitation-oriented search in finding the global best cipher text.

2. *Computational efficiency* : In (μ/ρ, λ) a lot of fitness calculation occurs in the intermediate steps of crossover as shown below. This has been eliminated in PRFRW selection scheme of parents.

A comparison of computations involved in fitness are listed below

- (μ/ρ, λ) selection employed in cryptography
  Minimum Number of times fitness is counted during encoding process = sample size* number of encryption iterations
  Minimum Number of times fitness is counted during decoding process = sample size* number of decryption iterations
- Deterministic and Pseudo-Deterministic GEA for cryptography
  Total Number of times fitness is counted during encoding process = number of encryption iterations
  Total Number of times fitness is counted during decoding process = number of decoding iterations

So we can infer that with PFRWR for the proposed cryptography there is a reduction in computation involved in deciding to choose parents for crossover operation. These observations emphasizes the efficiency of the proposed GEA for cryptography which is reliable and has a computational edge over cryptography with (μ/ρ, λ)-ES selection.

10.4 Conclusion

The chapter endeavors to couple the features and extended the benefits of synchronized chaotic systems with the non-deterministic class of evolutionary algorithms. The example produced is with a text but there is a scope to extend this scheme for image cryptography. Both the versions of the GEA for cryptography guarantees an initial population of unique candidate solutions. Closer observation throws light over the fact that a generation may evolve to repeat itself since the mixing of information is restricted within a fixed problem oriented solution space. Apart from the recombination schemes implemented, the proposed selection scheme of PFRWR can be tested by using other types of recombination schemes mainly uniform, heuristic and arithmetic. The proposed attempt is simple to implement and does not require great programming skills or complex formulation of the problem. Also apart from predetermining upon the population size and random crossover point, no other parameters need to be transmitted for encryption injecting further security into the system. The fitness measure is used to trigger the crossover operation with the subsequent crossovers being devised in a non greedy fashion, unbiased towards a fitter individual contesting to be a parent. This reduces considerable time, for the chromosomes are unsorted after the λ^{th} offspring is produced. Using PFRWR

selection in GEA for cryptography achieves exploration-oriented search by avoiding the sub optimal solutions and an exploitation-oriented search for seeking the global best. This interdisciplinary work between chaos and evolutionary theory is a preliminary attempt to apply this emerging area of research. Simulation result proves that the proposed GEA is effective and robust in employing to secure communication. It results in an enhanced convergence speed and correctly recovers the PT avoiding the problem of dominance of the super-fit individual with minimum computation as compared to the computation incurred in repeated fitness comparison in cryptography with $(\mu/\rho, \lambda)$ ES selection scheme.

Acknowledgements The author is thankful to S.Mukhopadhyay, University of Calgary, for some useful discussions.

References

1. Holland, J.H.: Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence. University of Michigan Press, Ann Arbor (1975)
2. Goldberg, D.E.: Commun. ACM **37**, 13 (1994)
3. Srinivas, M., Patnaik, L.M.: IEEE Comput. 17–26 (1994)
4. Fogel, L.J., Owens, A.J., Walsh, M.J.: Artificial Intelligence Through Simulated Evolution. Wiley, New York (1966)
5. Beyer, H.-G.: Evolution. Comput. **3**, 81 (1995)
6. Beyer, H.-G.: Natural Computing Series. Springer, Berlin (2001)
7. Beyer, H.-G., Schwefel, H.P.: Nat. Comput. **1**, 3 (2002)
8. Hansen, N., M*ü*ller, S.D., Koumoutsakos, P.: Evol. Comput. **11**, 1 (2003)
9. Hansen, N., Ostermeier, A.: Evol. Comput. **9**, 159 (2001)
10. Schwefel, H.P.: Evolution and Optimum Seeking. Wiley, New York (1995)
11. Mukhopadhyay, S., Mitra, M., Banerjee, S.: Chaos Synchronization and Cryptography for Secure Communications: Applications for Encryption, IGI Global, USA, pp. 476–509 (2010)
12. Al-Husainy, M.A.F.: Inform. Technol. J. **5**, 516 (2006)
13. Caponetto, R., Fortuna, L., Fazzino, S., Xibilia, M.G.: Evol. Comput. **7**, 289 (2003)
14. Banerjee, S., Rondoni, L., Mukhopadhyay, S., Misra, A.P.: Opt. Commun. **284**(9), 2278 (2011)
15. Banerjee, S., Rondoni, L., Mukhopadhyay, S.: Opt. Commun. **284**, 4623 (2011)
16. Mukhopadhyay, S., Banerjee, S.: Exp. Syst. Appl. **39**, 917 (2012)
17. Guo, S.M., Liu, K.T., Tsai, J.S.H., Shieh, L.S.: Comput. Math. Appl. **55**, 1225 (2008)
18. Wu, M.S., Lin, Y.L.: Genetic algorithm with a hybrid select mechanism for fractal image compression. Digit. Signal Process. **20**, 1150 (2010)
19. Yiqiang, G., Yanbin, W., Zhengshana, J., Jun, W., Luyane, Z.: Remote sensing image classification by the Chaos Genetic Algorithm in monitoring land use changes. Math. Comput. Model. **51**, 1408 (2010)
20. Klein, E., Mislovaty, R., Kanter, I., Kinzel, W.: Public-channel cryptography using chaos synchronization. Phys. Rev. E **016214**(1–4) (2005)
21. Banerjee, S., Saha, P., RoyChowdhury, A.: Phys. Scripta **63**, 177 (2001)
22. Mukherjee, P., Banerjee, S.: Phys. Scripta **82**, 055010 (2010)
23. Lin, W., He, Y.B.: Chaos **15**, 023705 (2005)
24. Lin, W.: Guanrong Chen Chaos **16**, 013134 (2006)
25. Banerjee, S.: Chaos Soliton Fractals **42**, 745 (2009)
26. Banerjee, S., Ghosh, D., Ray, A., Roy Chowdhury, A.: Europhys. Lett. **81**, 20006 (2008)
27. Ghosh, D., Banerjee, S., Roy Chowdhury, A.: Europhys. Lett. **80**, 30006 (2007)