Guillermo Huerta Cuéllar
Eric Campos Cantón
Esteban Tlelo-Cuautle *Editors*

Complex Systems and Their Applications

Second International Conference (EDIESCA 2021)

Springer

Complex Systems and Their Applications

Guillermo Huerta Cuéllar • Eric Campos Cantón •
Esteban Tlelo-Cuautle
Editors

Complex Systems and Their Applications

Second International Conference (EDIESCA 2021)

Springer

Editors
Guillermo Huerta Cuéllar
Universidad de Guadalajara
Centro Universitario de los Lagos
Lagos de Moreno, Jalisco, Mexico

Eric Campos Cantón
Instituto Potosino de Investigación
Científica y Tecnológica
San Luis Potosí, Mexico

Esteban Tlelo-Cuautle
Instituto Nacional de Astrofisica
Optica y Electronica
Tonantzintla, Puebla, Mexico

ISBN 978-3-031-02471-9 ISBN 978-3-031-02472-6 (eBook)
https://doi.org/10.1007/978-3-031-02472-6

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

This volume is intended for use by students, engineers and researchers interested in complex systems and their applications. The reader is assumed to have a basic knowledge of complex systems and dynamical nonlinear systems. EDIESCA is the Spanish acronym of Dissemination and Research in the Study of Complex Systems and their Applications. The first edition was held at Instituto Potosino de Investigación Científica y Tecnológica (IPICyT, México) in September 25–27, 2019, and the book of abstracts is available at https://sites.google.com/view/mesdia/sociedad-mexicana-de-sistemas-din%C3%A1micos-y-sus-aplicaciones/eventos/en-ediesca2021. The second edition of EDIESCA was held virtually during November 17–19, 2021, having as headquarters the facilities of the Los Lagos University Center in the city of Lagos de Moreno, Jalisco. The meeting encouraged the participation of researchers and students whose interests range from mathematical modeling to application of results in the area of complex systems, with an emphasis on chaos and dynamic systems. This book is the result of the presentations at EDIESCA 2021, integrating the work of students, research groups, and academic groups from related complex systems, as well as technological developments at national and international level. It is divided into five parts, covering the following topics: Part I, Synchronization, includes three chapters: ”Synchronization of Two Fiber Lasers with Optical Logarithmic Coupler: Experimental Implementation”; ”Anti-synchronization in a Pair of Coupled Multistable Systems”; and ”Analysis of Synchronizability in Small-World Complex Networks.” Part II, Cryptosystems, includes two chapters: ”Generation of Dynamical S-Boxes via Lag Time Chaotic Series for Cryptosystems” and ”Modification of the Quantum Logistic Map with Application in Pseudo-Random Bit Generation and Image Encryption.” Part III, Fractional Calculus, includes two chapters: ”On the Relationship Between Integer and Fractional PWL Systems with Multistable Behavior” and ”Approximation of Fractional-Order Controllers for Mechatronic Applications.” Part IV, Chaotic Systems, includes three chapters: ”Comparative Analysis of Chaotic Features of Maps Without Fixed Points”; ”A New 4-D Hyperchaotic System with No Balance Point, its Bifurcation Analysis, Multi-Stability, Circuit Simulation, and FPGA Realization”; and ”Displacement of

Equilibria and n-Double Wing Attractor Generation in the Piecewise Linearized Lorenz System." Part V, Applications, includes three Chapters: "Analysis of a Three-Dimensional Non-Autonomous Chaotic Circuit with a Thermistor as a Physical Memristor; "Inverse Filtering the Growth of Urban Sprawl with Cellular Automata Model"; and "A Metapopulation Network Model with Seasonal Succession to Analyze Dengue Disease in México. You are cordially invited to attend the following editions of EDIESCA. The third one will be held in 2022 at Universidad Autónoma de Baja California (UABC), Ensenada campus. Enjoy the book collection of Chapters of EDIESCA2021.

Lagos de Moreno, Mexico — Guillermo Huerta Cuellar
San Luis Potosí, Mexico — Eric Campos Cantón
Tonantzintla, Mexico — Esteban Tlelo Cuautle
November 2021

Organization

EDIESCA 2021 is locally organized by the Department of Exact Sciences and Technology at Universidad de Guadalajara Campus Centro Universitario de los Lagos, with support of Instituto Potosino de Investigación Científica y Tecnológica and Instituto Nacional de Astrofísica Óptica y Electrónica.

Executive Committee

Conference Chair,	Guillermo Huerta Cuellar (Universidad de
Program Chair,	Guadalajara, MX)
and Organizing Chair:	Eric Campos Cantón (Instituto Potosino de
	Investigación Científica y Tecnológica, MX)
	Esteban Tlelo Cuautle (Instituto Nacional
	de Astrofísica Óptica y Electrónica, MX)

Program Committee

Local Committee:	Guillermo Huerta Cuellar (Universidad of Guadalajara, MX)
	Rider Jaimes Reátegui (Universidad de Guadalajara, MX)
	Juan Hugo García López (Universidad de Guadalajara, MX)
Scientific Committee:	José Luis Echenausía Monroy (Centro de Investigación Cientifica y Educación Superior de Ensenada, MX)
	Héctor Eduardo Gilardi González (Universidad Panamericana, Aguascalientes, MX)
	Bahia Betzabet Cassal Quiroga (Universidad Panamericana, Aguascalientes, MX)

Session Chair:	Guillermo Huerta Cuellar (Universidad de Guadalajara, MX)
	Eric Campos Cantón (Instituto Potosino de Investigación Científica y Tecnológica, MX)
	Esteban Tlelo Cuautle (Instituto Nacional de Astrofíscia Óptica y Electrónica, MX)
	Rider Jaimes Reátegui (Universidad de Guadalajara, MX)
	Juan Hugo García López (Universidad de Guadalajara, MX)
	José Luis Echenausía Monroy (Centro de Investigación Cientifica y Educación Superior de Ensenada, MX)
	Héctor Eduardo Gilardi González (Universidad Panamericana, Aguascalientes, MX)
	Bahia Betzabet Cassal Quiroga (Universidad Panamericana, Aguascalientes), MX

Referees

A. Ruiz-Silva
A. Anzo-Hernández
B. B. Cassal-Quiroga
E. Campos
C. García-Grimaldo
E. Zambrano-Serrano
H. E. Giraldi Velázquez
J. H. García López
R. Jaimes-Reátegui
G. Huerta-Cuellar
J. L. Echenausía-Monroy
E. Jiménez-López
E. Tlelo-Cuautle
O. Guillen-Fernández
L. J. Ontañon-García
H. C. Rosu
R. J. Escalante-González
R. Rivera-Durón
J. Pena-Ramirez
J. G. Barajas-Ramirez
F. Ruiz-Oliveras
C. L. Pando-Lambruschini
I. Campos-Cantón
E. Inzunza-González

Sponsoring Institutions

Centro Universitario de los Lagos, Universidad de Guadalajara, Jalisco, México.
Instituto Potosino de Investigación Científica y Tecnológica, San Luis Potosí, México.
Instituto Nacional de Astrofísica Óptica y Electrónica, Puebla, México.

Contents

Part I
Synchronization

Chapter 1
Synchronization of Two Fiber Lasers with Optical Logarithmic Coupler: Experimental Implementation

J. O. Esqueda-de-la-Torre, J. H. García-López, G. Huerta-Cuellar, and R. Jaimes-Reategui

Abstract In this chapter, we present the study of the dynamical behavior of two periodic multistable erbium-doped fiber lasers (EDFL) unidirectionally coupled in "driver-driven configuration." The driver laser modulates the driven laser through the variation of a nonlinear optical logarithmic coupler. In order to determine the type of synchronization between both lasers, there are two predominant factors: the optical coupling between the driver and driven lasers; and the periodic behavior that each laser was previously set. Our results are presented through bifurcation diagrams, frequency locking, and synchronization error of time series of laser intensity signals between driver and driven EDFLs.

Keywords Multistability · Synchronization · Optical logarithmic coupler · Modulation · Bifurcation

1.1 Introduction

Erbium-doped fiber lasers (EDFL) have become an attractive optical device that generates chaos [1], periodic attractors [2], giant periodic pulses [3], and period doubling [4]. For this reason, many researchers have studied the dynamics of different experimental EDFL arrays, such as erbium-doped fiber dual-ring laser system [5], a chain of 980 nm-pumped erbium-doped fiber amplifiers (EDFA) [6], and tunable Q-switched EDFL [7]. In the last decades, the phenomenon of synchronization between coupled nonlinear systems and especially the ones with chaotic behavior has attracted the interest of the research community. The synchronization of nonlinear systems is an interesting phenomenon with a broad range of applications, such as in various complex physical, chemical, and biological

J. O. Esqueda-de-la-Torre (✉) · J. H. García-López · G. Huerta-Cuellar · R. Jaimes-Reategui
Optics and Complex Systems Laboratory, CULagos, Universidad de Guadalajara, Lagos de Moreno, Jalisco, Mexico
e-mail: jose.edelatorre@alumnos.udg.mx

G. Huerta Cuellar et al. (eds.), *Complex Systems and Their Applications*,
https://doi.org/10.1007/978-3-031-02472-6_1

systems [8–10] and secure and broadband communication system [11–13]. The concept of synchronization of two or more systems with chaotic behavior is the phenomenon in which the coupled systems can adjust their dynamics to a common behavior (equal trajectories or phase locking), due to forcing or coupling [14]. Synchronization theory has been studied since the 1980s and early 1990s [15, 16]. Since then, a great number of research works based on synchronization of nonlinear systems have risen, and many different synchronization schemes depending on the nature of the coupling schemes and of the interacting systems have been presented. Complete or full chaotic synchronization [17], phase synchronization [18], lag synchronization [19], generalized synchronization [20], anti-phase synchronization [21], and anticipating synchronization [22] are the most interesting types of synchronization. Motivated by the last concepts mentioned previously, this chapter presents a "driver-driven system configuration," to study the dynamics behavior, as the phase and amplitude synchronization, of two unidirectionally coupled EDFLs. Considering the obtained results, we found that there are some predominant factors: the first, the optical coupling strength (optical attenuator) of the driver laser signal applied to the driven laser and second, the specific dynamics of each laser is previously set. The rest of the work is distributed as follows: In Sect. 1.2, the dynamical behavior of an erbium-doped fiber laser (EDFL) is presented by means of time series and bifurcation diagrams. The experimental implementation of the "driver-driven configuration" is described throughout Sect. 1.3. The implemented methodology for the system characterization, as well as the obtained results, is given in Sect. 1.4. The main conclusions of our work are stated in the last section of the chapter.

1.2 Dynamics of an Erbium-Doped Fiber Laser (EDFL)

In order to obtain the uncoupled EDFL behavior obtained by numerical simulations, the equations used are shown in Eqs. 1.1, 1.2, and 1.3, where x is the laser intensity, y is the population inversion, t is the time, and parameters a, b, c, and d are the constant values that are given in Table 1.1. P_{pump} is the pump harmonic modulation, F_m is the frequency modulation, and A_m is the amplitude modulation [23].

$$\frac{dx}{dt} = axy - bx + c(y + 0.3075) \tag{1.1}$$

$$\frac{dy}{dt} = (dxy - (y + 0.3075) + P_{pump}(1 - e^{-18(\frac{1-(y+0.3075)}{0.6150})}) \tag{1.2}$$

$$P_{pump} = 506[1 + A_m \sin(2\pi F_m t)]. \tag{1.3}$$

Table 1.1 Constant values of Eqs. 1.1, 1.2, and 1.3 [24]

Parameters	Values
a	6.6207×10^7
b	7.4151×10^6
c	0.0163
d	4.0763×10^3
A_m	1

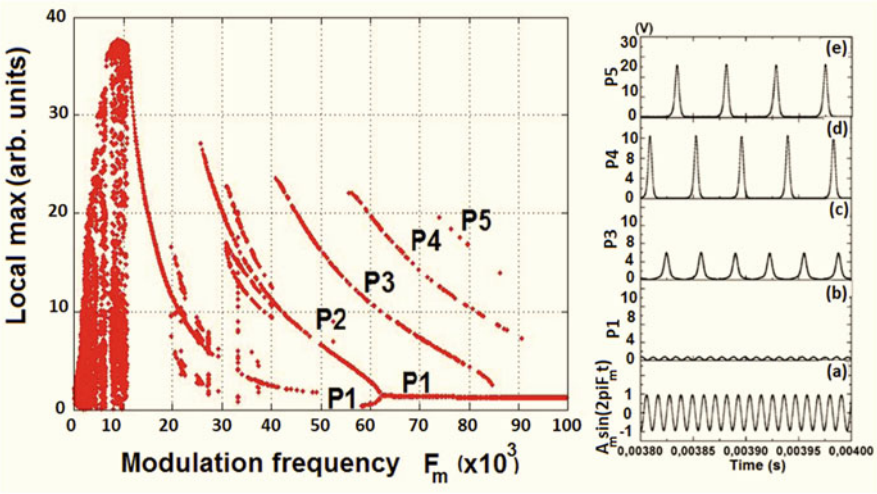


Fig. 1.1 (Left panel) EDFL bifurcation diagram of Eqs. 1.1 and 1.2, where $A_m = 1$ and $0 < F_m < 100$ kHz and (right panel) (**a**) modulation frequency and periodic attractors of (**b**) P1, (**c**) P3, (**d**) P4, and (**e**) P5 in the EDFL for $A_m = 1$ and $F_m = 80$ kHz [24]

1.2.1 Bifurcation Diagram and Periodic Attractors

Based on the EDFL equations model (Eqs. 1.1, 1.2, and 1.3), Fig. 1.1 shows the bifurcation diagram of local maxima of the laser intensity (x) as a function of modulation frequency (F_m) for fixed modulation amplitude ($A_m = 1$).

We can observe in Fig. 1.1(left panel) the coexistence of 4 different periodic attractors of P1, P3, P4, and P5 when the frequency modulation and the amplitude modulation are set up to $F_m = 80$ kHz, and $A_m = 1$, respectively. While the different periodic behaviors in this system are shown in Fig. 1.1b–e (right panel), there are clearly differences between each periodic attractor, the amplitude, and the frequency, but they are all proportional to the modulation frequency F_m. It should be noted that in Fig. 1.1a, b the frequency response of the P1 attractor is the same as the modulation frequency F_m. In Fig. 1.1c, the frequency response of P3 attractor is $\frac{1}{3}$ of modulation frequency F_m. In Fig. 1.1d, the frequency response of P4 attractor is $\frac{1}{4}$ of modulation frequency F_m. Finally, in Fig. 1.1e, the frequency response of the P5 attractor is $\frac{1}{5}$ of modulation frequency F_m [24].

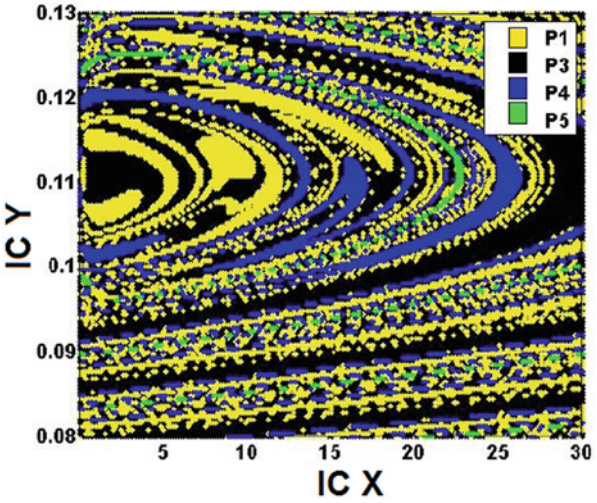


Fig. 1.2 Basin of attraction of EDFL when $A_m = 1$ and $F_m = 80\,\text{kHz}$ [24]

1.2.2 Basin of Attraction

The periodic attractors exhibited by this EDFL depend on the change of the initial conditions (IC). The basin of attraction is constructed by changing the IC. Figure 1.2 shows the basin of attraction of equation system (Eqs. 1.1 and 1.2), where the yellow color corresponds to period 1 (P1), black color corresponds to period 3 (P3), blue color corresponds to period 4 (P4), and green color corresponds to period 5 (P5).

1.3 Experimental Setup and Bifurcation Diagrams

1.3.1 Experimental Setup

The experimental setup is illustrated in Fig. 1.3. It consists of two coupled EDFL lasers, a driver, and a driven EDFL. The driver EDFL consists of a 6.5 m laser cavity formed by an active 1.5 m heavily erbium-doped single-mode fiber (EDF1) with a 2.7 μm core diameter, a wavelength-divider multiplexing coupler (WDM1-WD9850FD), and two fiber Bragg gratings (FBG1 and FBG2) with 0.288 and 0.544 nm HMFW with reflectiveness of 100 and 95.88%, respectively, at a 1550 nm wavelength. All optical components are connected by a single-mode fiber $SMF-28$ with a 200 μm cladding diameter. The EDFL driver is pumped with a 977 nm laser diode (LD1-BL976PAG500) and pumps a wavelength division multiplexer (WDM1).

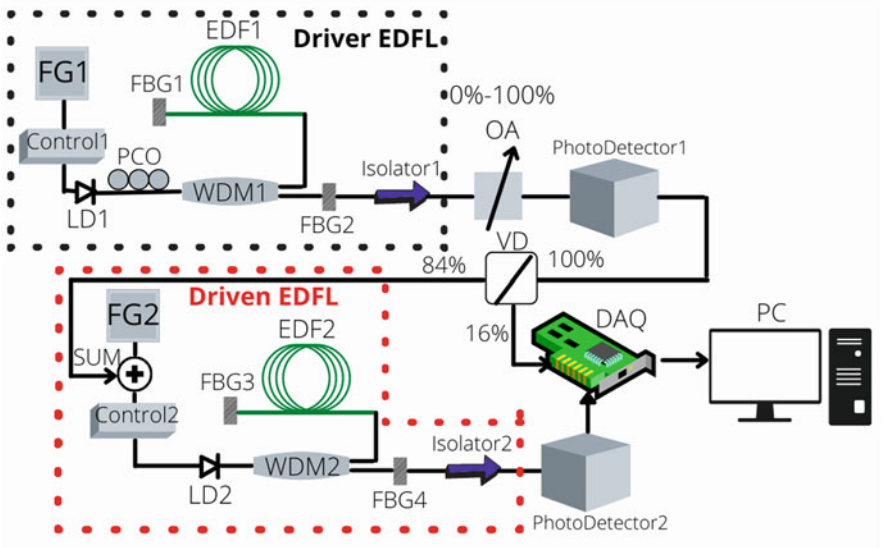


Fig. 1.3 Experimental setup of two coupled EDFLs where FG1 is the wave function generator, control1 and control2 are the driver currents, LD1 and LD2 are the pump laser diodes, PCO is a polarize controller, WDM1 and WDM2 are the wavelength divisor multiplexors, FBG1, FBG2, FBG3, and FBG4 are the Bragg gratings, EDF1 and EDF2 are the erbium-doped fibers, Isolator1 and Isolator2 are the optical isolators, OA is the optical attenuator, PhotoDetector1 and PhoteDetector2 are the photodiodes, VD is a voltage divider, SUM is an electrical signal combiner, DAQ is a data acquisition card, and PC is a personal computer

The parameters of the driven EDFL differ from the driver EDFL. The driven EDFL has a 5.8 m laser cavity formed by a 1.5 m active heavily erbium-doped fiber (EDF2) with a 2.7 μm core diameter, a wavelength-divider multiplexing coupler (WDM2-WD9 850FD), and two fiber Bragg gratings (FBG3 and FBG4) with 0.288 and 0.544 nm HMWF with reflectiveness of 100 and 96.4%, respectively, at a 1550 nm wavelength. All optical components are connected by a single-mode fiber $SMF-28$ with a 200 μm cladding diameter. The driven EDFL is pumped by a 977 nm laser diode (LD2-BL976PAG500) and pumps a wavelength division multiplexer (WDM2). The driver EDFL is pumped by a 977 nm laser diode (LD1-BL976PAG500) through a polarize controller (PCO) and pumps the driven EDFL via a wavelength-divider multiplexer (WDM1). The diode-pumped laser is controlled by a laser diode controller (Control1-LDC-ITC510). In our experiments, the diode current is fixed at 118.4 mA that corresponds to a 20 mW pump power, while the lasing threshold of the driver EDLF is 110 mA. To modulate the driver EDFL, a harmonic signal $A_m \sin(2\pi F_m t)$ is applied to the diode-pumped current from a function generator (FG1-AFG3102). The experimental data are obtained by a data acquisition card (DAQ) and stored in the computer (PC).

The optical output signal from the fiber Bragg gratings (FBG2) passes through an isolator (Isolator1) and then enters to an optical attenuator (OA-PW410), which attenuates the signal by a factor k, where $k = 0\%$ means zero optical transmittance and $k = 100\%$ means full optical transmittance. After passing through the optical attenuator, the optical signal enters a photodetector (PhotoDetector1) connected to a voltage divider (VD) that has two outputs; the first yields 84% of the electrical signal, while the second provides 16% of the electrical signal. The 84% output of the driver EDFL is connected to the laser diode current controller (Control2-LD-ITC510) (which produces a harmonic modulation $A_m \sin(2\pi F_m t)$ of driven EDFL in order to couple both EDFLs through a summing electrical device. (SUM). Finally, the 16 and 100% output electrical signals from PhotoDetector1 and PhotoDetector2, respectively, are connected to a data acquisition card (DAQ) and stored in a personal computer (PC) to analyze the dynamics of both lasers.

The objective of this chapter is to study which kind of synchronization occurs in the driver-driven experimental setup by changing the signal optical coupling from the driver EDFL intensity to the driven EDFL for different periodic attractors.

1.3.2 Optical Attenuator (OA)

The coupling between driver and driven EDFLs is done through an optical logarithmic attenuator (OA). We have used a passive single-mode EigenLight power monitor 410 as the optical attenuator (OA). This is a 40 dB variable attenuator with a power range of -50 to $+16$ dBm. It has a screw to control the output power. The number of turns on the screw gives an exponential output (see Fig. 1.4). To measure the optical output power of the optical attenuator (OA), we have used a 0.84 V signal. The optical attenuator screw has been divided into 7 and a half turns, and each turn is divided into 12 parts having 90 steps in total (exponential behavior in Fig. 1.4, blue points). The percentage rate of the number of turns on the screw is shown in Table 1.2 that corresponds to the red straight line in Fig. 1.4.

The 32 attenuation values for each corresponding optical transmittance percentage k are shown in the next table.

For the experimental realization, the optical attenuation of the optical driver signal entering into the driven EDFL is controlled by the number of turns on the screw, for which we defined 32 values. Table 1.2 shows the 32 values defined to control the optical attenuation; the value 32 corresponds to the 100% of driver EDFL signal, and it begins to diminish for every twelfth of a turn on the screw, until it has the 1% equivalent to 7 $\frac{6}{12}$ turn.

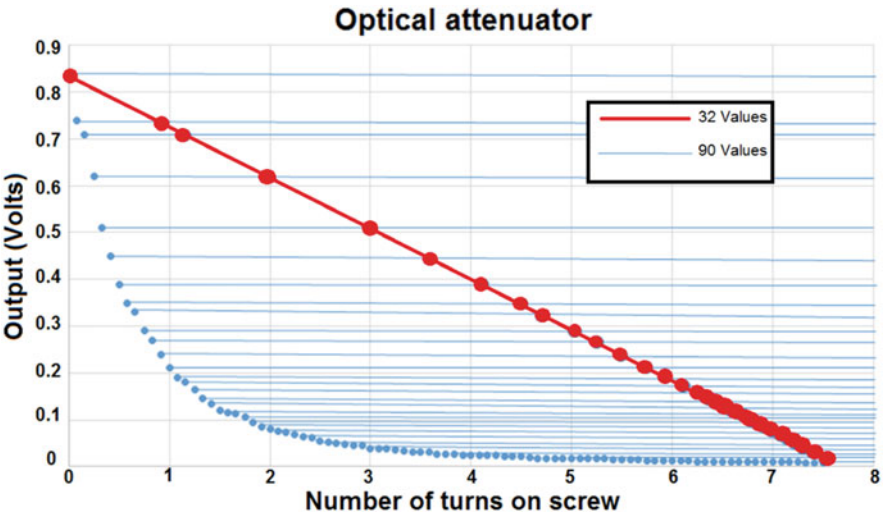


Fig. 1.4 Linearized optical attenuator (OA) for 32 different attenuation values

Table 1.2 The 32 linear values of logarithmic attenuator

Value	k	Turns	Value	k	Turns	Value	k	Turns	Value	k	Turns
1	1%	$7\frac{6}{12}$	9	9%	$2\frac{1}{12}$	17	20%	$1\frac{9}{12}$	25	39%	$\frac{7}{12}$
2	2%	5	10	10%	$1\frac{2}{12}$	18	21%	$1\frac{10}{12}$	26	42%	$\frac{6}{12}$
3	3%	$3\frac{10}{12}$	11	11%	$1\frac{3}{12}$	19	23%	$1\frac{11}{12}$	27	46%	$\frac{5}{12}$
4	4%	$3\frac{3}{12}$	12	12%	$1\frac{4}{12}$	20	25%	1	28	53%	$\frac{4}{12}$
5	5%	$2\frac{11}{12}$	13	13%	$1\frac{5}{12}$	21	28%	$\frac{11}{12}$	29	60%	$\frac{3}{12}$
6	6%	$2\frac{7}{12}$	14	14%	$1\frac{6}{12}$	22	32%	$\frac{10}{12}$	30	74%	$\frac{2}{12}$
7	7%	$2\frac{5}{12}$	15	16%	$1\frac{7}{12}$	23	34%	$\frac{9}{12}$	31	84%	$\frac{1}{12}$
8	8%	$2\frac{3}{12}$	16	17%	$1\frac{8}{12}$	24	36%	$\frac{8}{12}$	32	100%	0

Based on the previous characterization mentioned, the driven EDFL pump power is given by the following equation:

$$P_{PumpDriven} = 506[1 + A_m \sin(2\pi F_m t + kx], \tag{1.4}$$

where k is the optical transmittance and x is the laser intensity from driver EDFL.

1.3.3 Experimental Bifurcation Diagram of the Driver EDFL

Figure 1.5 illustrates the bifurcation diagram of the isolated driver EDFL peak intensity as a function of the driving frequency F_m for $A_m = 1$ V. The bifurcation

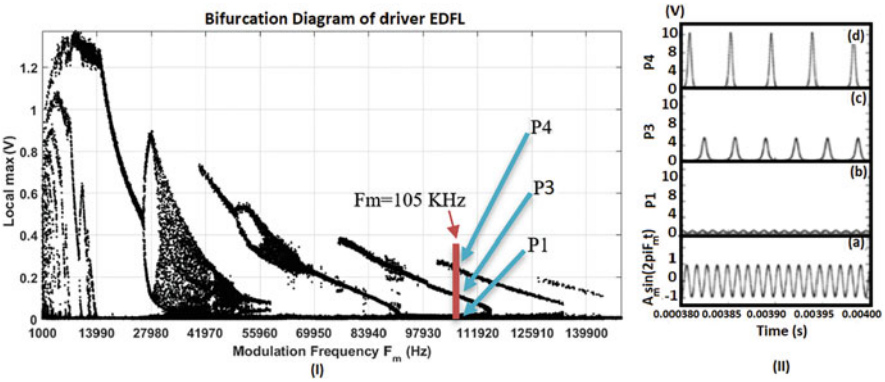


Fig. 1.5 (I) Experimental bifurcation diagram of peak intensity (in V) of driver EDFL as a function of modulation frequency F_m for fixed $A_m = 1$ V. For fixed $F_m = 105$ kHz and $A_m = 1$ V, the driver EDFL exhibits the coexistence of three periodic attractors (II): (**b**) P1, (**c**) P3, and (**d**) P4 and finally, (**a**) harmonic modulation $A_m \sin(2\pi F_m t)$ for $F_m = 105$ kHz and $A_m = 1$ V

diagram has been obtained by switching on and off the function generator (FG1) 15 times each 100 Hz in a range of 1 kHz$< F_m <$ 150 kHz, so that the initial conditions are randomly changed. One can distinguish branches corresponding to different periodic regimes that are born in saddle-node bifurcations as the control parameter F_m increases. For certain ranges of the modulation frequency, several attractors coexist. For example, in the range of 104 kHz$< F_m <$ 113 kHz, P1, P3, and P4 coexist, and in the range of 126 kHz$< F_m <$ 131 kHz, P1, P4, and P5 coexist. In this chapter, we are interested in studying the lasers synchronization for coexistent attractors of P1, P3, and P4 that were obtained for $A_m = 1$ V and $F_m = 105$ kHz.

The bifurcation diagram was obtained by measuring the local maximum of the time series registered by the data acquisition card (DAQ) from the photodetector (PhotoDetector1).

1.3.4 Experimental Bifurcation Diagram of the Driven EDFL

Figure 1.6 illustrates the bifurcation diagram of the isolated driven EDFL peak intensity from the time series obtained by PhotoDetector2 as a function of the modulation frequency F_m for $A_m = 1$ V. Using the same procedure as in Fig. 1.5, the wave function generator (FG2) was switched on and off 15 times for each 100 kHz of F_m in a range of 0 Hz$< F_m <$ 115 kHz, to change the initial conditions. Just as in Fig. 1.5, in Fig. 1.6, we can distinguish branches corresponding to different periodic regimes that are born in saddle-node bifurcations as the control parameter F_m increases. For certain ranges of the modulation frequency, several attractors

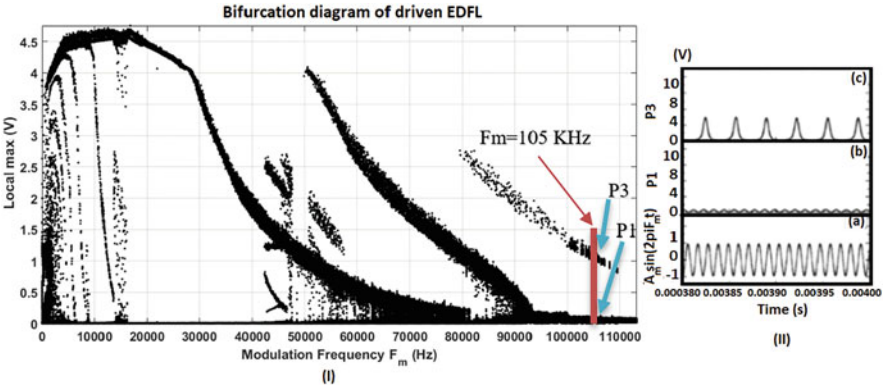


Fig. 1.6 (I) Experimental bifurcation diagram of V of the driven EDFL as a function of the modulation frequency F_m for fixed $A_m = 1$ V. For fixed $F_m = 105$ kHz and $A_m = 1$ V, the driven EDFL exhibits the coexistence of two periodic attractors (II): (**b**) P1 and (**c**) P3, and finally, (**a**) harmonic modulation $A_m \sin(2\pi F_m t)$

coexist. For example, in the range of 50 kHz$< F_m <$ 80 kHz, P1 and P2 coexist, and in the range of 80 kHz$< F_m <$ 110 kHz, P1 and P3 coexist.

1.4 Results

1.4.1 Bifurcation Diagrams of Laser Intensity of Driver and Driven EDFLs

Figures 1.7, 1.8, and 1.9 show the bifurcation diagrams of peak intensity in volts (V) of the driver and driven EDFLs coupled as a function of the optical transmittance (k) (see the experimental setup in Fig. 1.3). The modulation frequency and the amplitude of the driver EDFL have been fixed $F_m = 105$ kHz and $A_m = 1$ V, respectively, in order to have the driver EDFL exhibited in a multistable behavior of period P1 Fig. 1.7a, c, period P3 Fig. 1.8a, c, and period P4 Fig. 1.9a, c. While the driven EDFL has been set up in the two coexistent periodic attractors: period P1 Figs. 1.7b, 1.8b, and 1.9b; and period P3 Figs. 1.7d, 1.8d, and 1.9d.

Driver EDFL in P1 Figure 1.7a, b shows the results of the coupling between the driver EDFL in P1 that modulates the driven EDFL in P1 and P3, respectively. In this figure, we can see resistance from the driven EDFL to follow the behavior of the driver EDFL, i.e., the coupling between driver and driven EDFL only produces destabilization in the last laser, which is observed in all ranges of the optical transmittance k. While in Fig. 1.7c, d, when the driver EDFL is in P1 and the driven EDFL is in P3, respectively, the result of this coupling between both lasers results in that the driven EDFL keeps its periodic behavior of P3 with a short increase in amplitude of its time series.

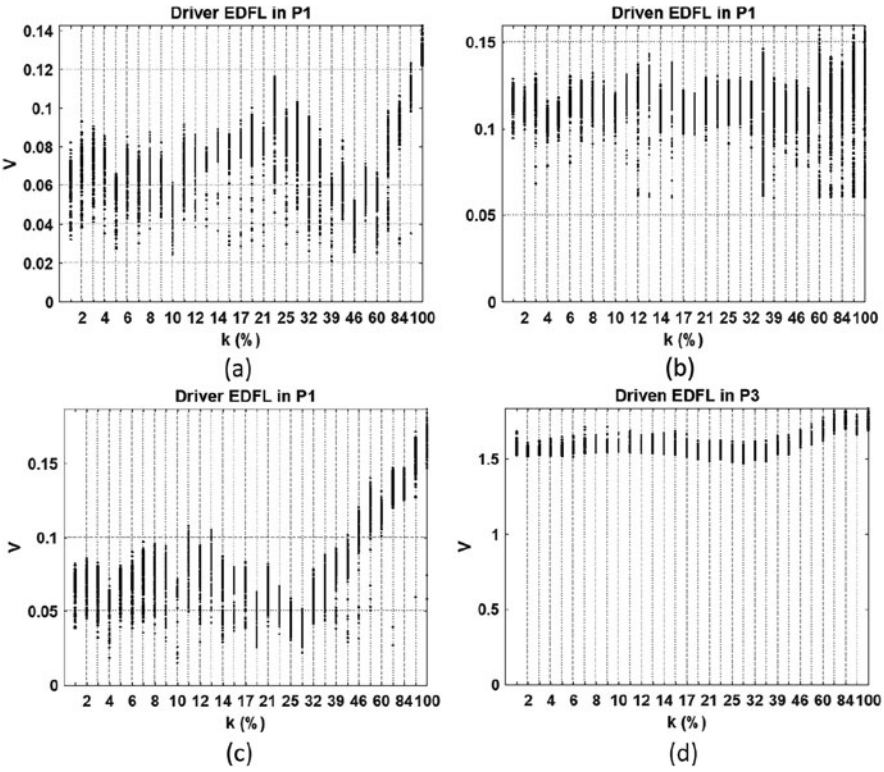


Fig. 1.7 Bifurcation diagrams of V as a function of optical transmittance k for (**a**) driver EDFL in P1 and (**b**) driven EDFL in P1, (**c**) driver EDFL in P1, and (**d**) driven EDFL in P3

Driver EDFL in P3 In a similar way to that in Figs. 1.7 and 1.8a, b, the driver EDFL is fixed in P3 and the driven EDFL is fixed in P1, respectively. In Fig. 1.8b, we can see a threshold value of the optical transmittance $k = 60\%$. For $k < 60\%$, the driven EDFL keeps its P1 behavior, and for $k > 60\%$, the driven EDFL switches to P3 behavior, i.e., the driven EDFL follows the driver EDFL behavior producing synchronization between the coupled lasers. Likewise, in Fig. 1.8d, we can identify two threshold values of the optical transmittance: $k = 11\%$ and $k = 74\%$. For $k < 11\%$, the driven EDFL keeps its periodic behavior. While in the interval of optical transmittance $11\% < k < 60\%$, the driven EDFL switches into a P1 and further $k > 60\%$, and the driven EDFL switches into P3 with an increase of the amplitude of its time series. The last result demonstrates that the driven EDFL follows the driver EDFL behavior.

Driver EDFL in P4 In the same way as in Fig. 1.8a, b, in Fig. 1.9a, b, when $k > 46\%$, the driven EDFL switches from P1 into P4 (the same behavior as the driver EDFL) producing synchronization between the coupled lasers. While in Fig. 1.9c,

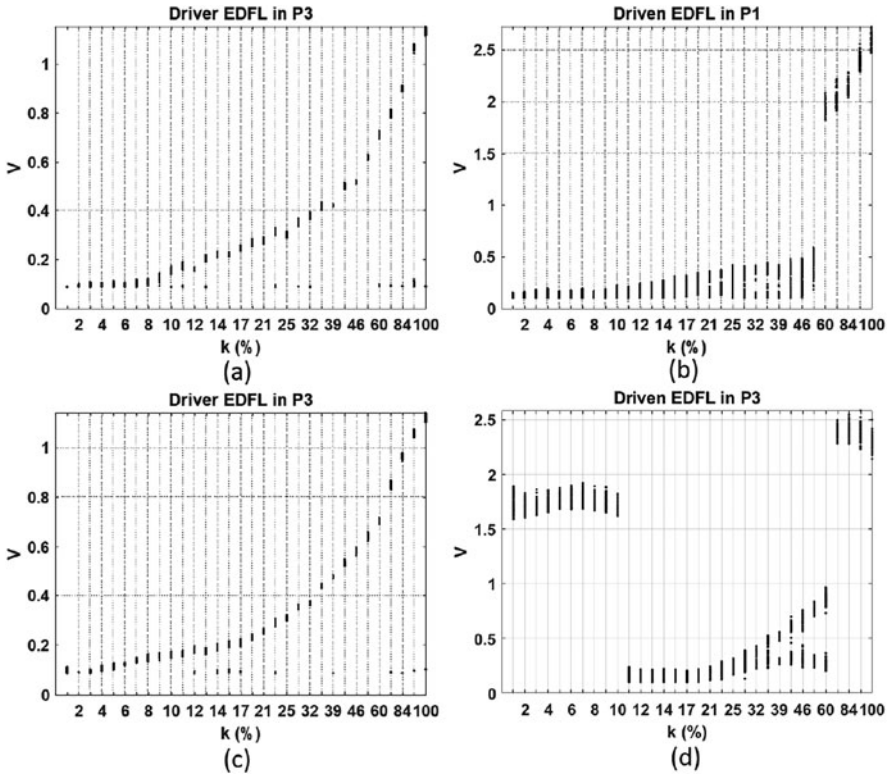


Fig. 1.8 Bifurcation diagrams of peak intensity (in V) as a function of optical transmittance k for (**a**) driver EDFL in P3, (**b**) driven EDFL in P1, (**c**) driver EDFL in P3, and (**d**) driven EDFL in P3

d, again we can observe two threshold values of the optical transmittance k: $k = 3\%$ and $k = 53\%$. The driven EDFL keeps the P3 behavior for $k \leq 3\%$, and for an increase of optical transmittance in the interval $3\% < k < 53\%$, the driven EDFL switches into P1. Furthermore, for $k \geq 53\%$, the synchronization between the two coupled lasers is produced, i.e., driven and driver EDFL are in P4.

1.4.2 Frequency Difference Between Driver and Driven EDFLs of Principal Component Domain of Fourier Transform

The frequency responses of these experimental lasers depend crucially on the initial conditions they have been previously set up (as seen in Sect. 1.2.1). The modulation frequency and amplitude have been chosen to be the same ($F_m = 105\,\text{kHz}$ and

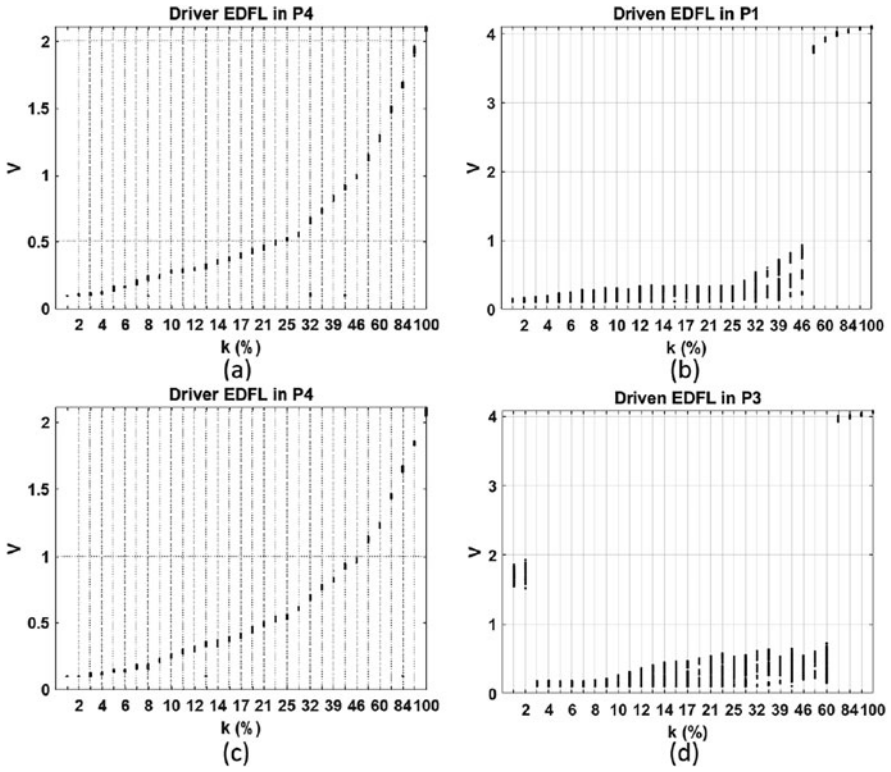


Fig. 1.9 Bifurcation diagrams of peak intensity (in V) as a function of the optical transmittance k for (**a**) driver EDFL in P4, (**b**) driven EDFL in P1, (**c**) driver EDFL in P4, and (**d**) driven EDFL in P3

$A_m = 1\,\text{V}$) for both driver and driven EDFLs, though the driver EDFL shows a multistable regime (coexistence of 3 or more periodic attractors) and the driven EDFL shows a bistable regime (coexistence of 2 periodic attractors). Having $F_m = 105\,\text{kHz}$, the frequency responses of these lasers depend on their attractor; for instance, the frequency response of periodic attractor P1 is the same as the modulation frequency F_m ($\frac{F_m}{1} = 105$) kHz, the frequency response of periodic attractor of P3 is a third of the modulation frequency F_m ($\frac{F_m}{3} = 35\,\text{kHz}$), and the frequency response of periodic attractor of P4 is a fourth of the modulation frequency F_m ($\frac{F_m}{4} = 26250\,\text{Hz}$). These lasers have been coupled for different configurations even though they have different frequency responses. The following results show the difference of the principal frequency of Fourier transform of driver and driven EDFLs. In most cases when they are frequency-synchronized, this difference tends to zero.

Phase Synchronization In order to determine the phase synchronization between the driver and driven EDFLs, in Fig. 1.10, we have measured the frequency

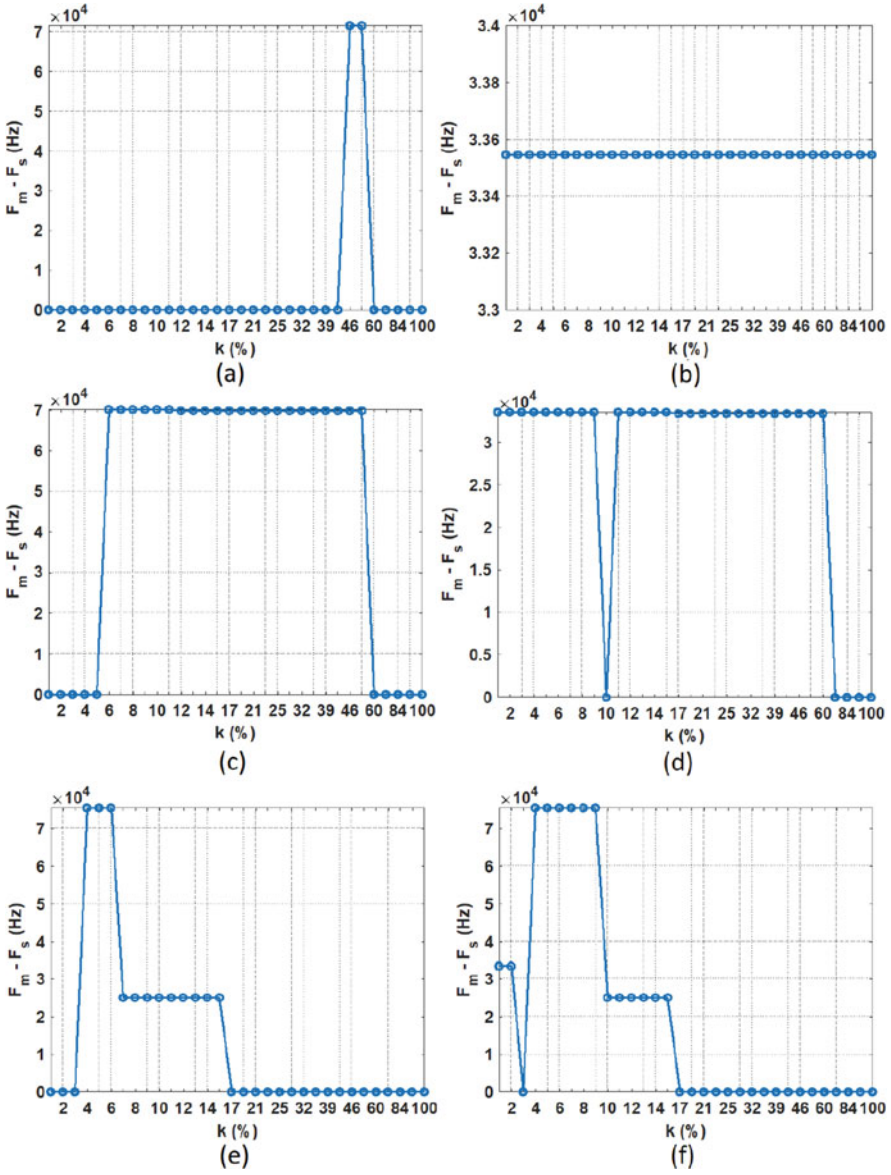


Fig. 1.10 Frequency difference of principal component frequency of the Fourier transform between the driver and driven EDFLs when (**a**) driver and driven EDFLs are in P1, (**b**) driver EDFL in P1 and driven EDFL in P3, (**c**) driver EDFL in P3 and driven EDFL in P1, (**d**) driver and driven EDFLs in P3, (**e**) driver EDFL in P4 and driven EDFL in P1, and (**f**) driver EDFL in P4 and driven EDFL in P3

difference $\Delta F = F_1 - F_2$, where F_1 and F_2 are the dominant spectral frequencies of the driver and driven EDFLs' time series, respectively, as a function of k. Before the increase of optical transmittance k, the dynamics of the driver and driven EDFLs were fixed: Fig. 1.10a, driver in P1 and driven in P1, Fig. 1.10b, driver in P1 and driven in P3, Fig. 1.10c, driver in P3 and driven in P1, Fig. 1.10d, driver in P3 and driven in P3, Fig. 1.10e, driver in P4 and driven in P1, and Fig. 1.10f, driver in P4 and driven in P3.

In Fig. 1.10e, we can distinguish several intervals of k values where the ΔF evolution can be observed. In the interval $0\% < k \leq 3\%$, due to the small value of k between the driver and driven EDFLs, the ΔF cannot be distinguished, see time series and power spectrum in Fig. 1.11a, b, respectively. For the interval $4\% < k \leq 6\%$, we have that $\Delta F = 80\,\text{kHz}$, and the details of the time series and power spectrums are shown in Fig. 1.11c, d. For $7\% < k \leq 16\%$, $\Delta F = 26.2\,\text{kHz}$, and the time series and power spectrums are shown in Fig. 1.11e, f. Furthermore, in the interval $17\% < k \leq 100\%$, $\Delta F = 0$, and the time series and power spectrums are shown in Fig. 1.11g, h. The first three k intervals represent frequency locking, and the last interval represents phase synchronization between the driver and driven systems. A similar procedure was developed in Fig. 1.10a–d and f. The phase synchronization relies on the periods where the driver and driven EDFLs were fixed. For instance, phase synchronization appears when: $k = 60\%$ for Fig. 1.10a, $k = 1\%$ for Fig. 1.10b, $k = 60\%$ for Fig. 1.10c, $k = 74\%$ for Fig. 1.10d, $k = 17\%$ for Fig. 1.10e, and $k = 17\%$ for Fig. 1.10f. Likewise, when the driver and driven EDFLs are maintained in the same period, for small values of k, phase locking is achieved or $\Delta F = 0$.

1.4.3 Synchronization Error Estimation

The following results show the analysis of the synchronization error estimation in order to identify how similar or different are the signals of the driver and driven EDFLs. The following equation is used to obtain the synchronization error, where driverEDFL and drivenEDFL represent the peak intensity V of time series from driver and driven EDFLs as a function of k, respectively [15–17].

$$< e >= \overline{\sqrt{(driverEDFL - drivenEDFL)^2}}. \tag{1.5}$$

Figure 1.12a–f shows the average synchronization error $< e >$ between the driver and driven EDFLs as a function of optical transmittance k when the behavior of the driver and driven EDFLs was fixed. In Fig. 1.12a, both EDFLs were fixed to P1, the $< e >$ has small values for $k < 7\%$, while for $k > 7\%$, the $< e >$ increases. In Fig. 1.12b, the driver EDFL is in P1 and the driven EDFL is in P3, and the $< e >$ decreases as k increases with some leaps; in Fig. 1.12c, the driver EDFL is in P3 and the driven EDFL is in P1, and the $< e >$ maintains a constant value close to $0.07V$

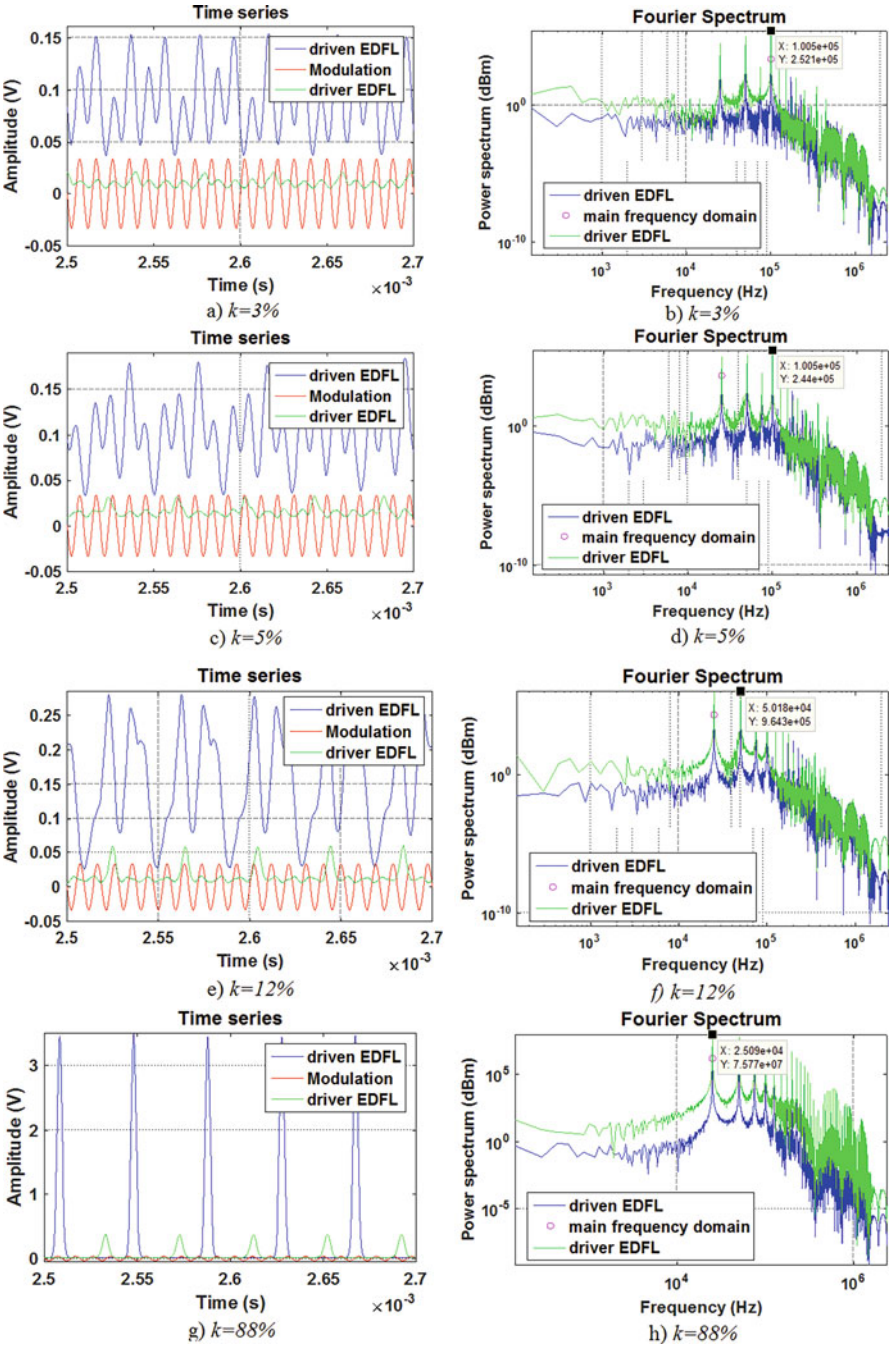


Fig. 1.11 Time series and power spectrums of driver and driven EDFLs in P4 and P1, respectively; (**a**) time series and (**b**) power spectrum for $k = 3\%$, (**c**) time series and (**d**) power spectrums for $k = 5\%$, (**e**) time series and (**f**) power spectrums for $k = 12\%$, and (**g**) time series and (**h**) power spectrums for $k = 88\%$

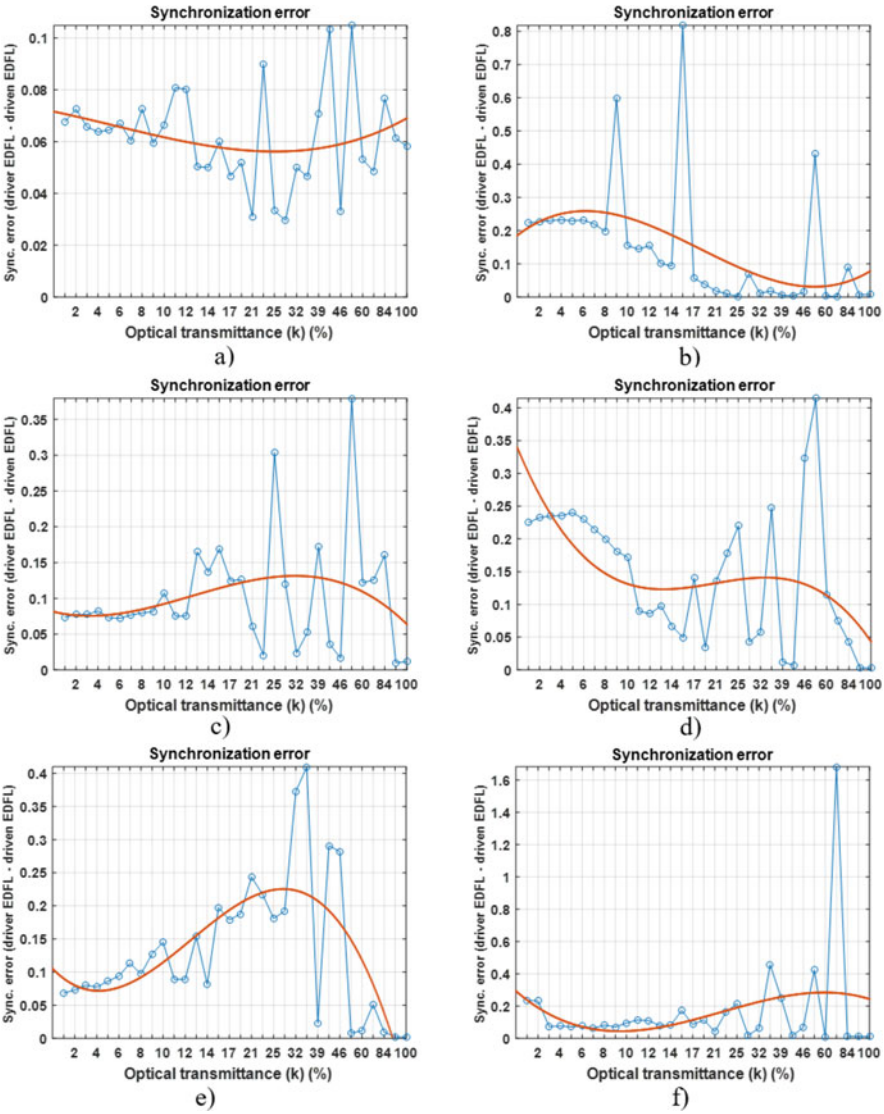


Fig. 1.12 Synchronization error estimation when (**a**) driver and driven EDFLs are in P1, (**b**) driver EDFL is in P1 and driven EDFL is in P3, (**c**) driver EDFL is in P3 and driven EDFL is set in P1, (**d**) driver and driven EDFL are in P3, (**e**) driver EDFL is in P4 and driven EDFL is in P1, and (**f**) driver EDFL is in P4 and driven EDFL is in P3. In figures (**a**)–(**f**), the orange curve represents the best fit curve and gives us the synchronization error tendency as a function of k

for $k < 9\%$, and for $k > 9\%$, we have different leaps. In Fig. 1.12d, both EDFLs are in P3, similar to Fig. 1.12b, and the $< e >$ decreases with some leaps as k increases. In Fig. 1.12e, the driver EDFL is in P4 and the driven EDFL is in P1, and $< e >$ increases for $1\% < k < 34\%$ with some leaps, and for $k > 34\%$, $< e >$ decreases considerably. Finally, in Fig. 1.12f, the driver EDFL is in P4 and the driven EDFL is in P3, and the $< e >$ maintains a constant value close to $0.1V$ for $k < 14\%$, and for $k > 14\%$, we have some leaps until $< e >$ tends to zero for $k > 84\%$. From these figures, we can see that the driven EDFL does not necessarily follow the signal of the driver EDFL, and the mean synchronization error shows irregular patterns. In Fig. 1.12a–f, the orange curve represents the best fit curve and gives us the synchronization error tendency as a function of k.

1.5 Conclusions

In this chapter, an experimental study of the dynamics behavior of two periodic multistable EDFLs unidirectionally coupled in driver-driven configuration was realized. The driver EDFL modulates the pump current of the driven EDFL through the variation of a nonlinear optical coupler (optical logarithmic attenuator). First, the system dynamics has been analyzed by constructing the bifurcation diagrams of local maximum of time series of the driver and driven laser intensity without coupling to identify multistable and bistable regimes by switching the initial conditions of the modulation frequency applied to the pump current control of each laser. The modulation frequency and amplitude were fixed at $F_m = 105\,\text{kHz}$ and $A_m = 1\,\text{V}$ for each laser, where driver and driven EDFLs show the coexistence of periodic attractors: P1, P3, and P4 for the driver EDFL; and P1 and P3 for the driven EDFL. Second, the coupling between the driver and driven system in their different coexistence states shows interesting dynamical behaviors, one of them is referred to the incapacity of the driver EDFL to control the driven EDFL (Fig. 1.7), i.e., driver and driven are in P1, and a non-periodic behavior is showed by the driven EDFL. In addition, the driven EDFL does not change its behavior of P3 when it is modulated by the driver EDFL in P1. Likewise, when the driver EDFL is fixed in either P3 or P4, we can distinguish a k threshold for which both lasers get synchronized. Phase synchronization phenomena were also achieved for different locking frequencies. Some exceptions were found. The average synchronization error $< e >$ reaches small values for high k in most of the studied cases. In general, we experimentally demonstrated that it is possible to synchronize different real multistable EDFL systems since defined configurations.

References

1. Y. Imai, H. Murakawa, T. Imoto, Chaos synchronization characteristics in erbium-doped fiber laser systems. Opt. Commun. **217**(1–6), 415-420 (2003). https://doi.org/10.1364/JOSAB.22.002107
2. A.N. Pisarchik, A.V. Kiryanov, Y.O. Barmenkov, R. Jaimes-Reátegui, Dynamics of an erbium-doped fiber laser with pump modulation: theory and experiment. J. Opt. Soc. Am B **22**(10), 2107–2114. (2005) https://doi.org/10.1016/S0030-4018(03)01122-2
3. R. Jaimes-Reátegui, J.O.E. De La Torre, J.H. García-López, G. Huerta-Cuellar, V. Aboites, A.N. Pisarchik, Generation of giant periodic pulses in the array of erbium-doped fiber lasers by controlling multistability. Opt. Commun. **477**, 126355 (2020). https://doi.org/10.1016/j.optcom.2020.126355
4. E. Lacot, F. Stoeckel, M. Chenevier, Dynamics of an erbium-doped fiber laser. Phys. Rev. A **49**(5), 3997 (1994). https://doi.org/10.1103/PhysRevA.49.3997
5. K. Shen, R. Wang, Synchronization of chaotic systems modulated by another chaotic system in an erbium-doped fiber dual-ring laser system. IEEE J. Quant. Electron. **37**(8), 960–963 (2001). https://doi.org/10.1109/3.937385
6. R.G. Smart, J.W. Sulhoff, J.L. Zyskind, J.A. Nagel, D.J. DiGiovanni, Gain peaking in a chain of 980 nm-pumped erbium-doped fiber amplifiers. IEEE Photon. Technol. Lett. **6**(3), 380–382 (1994). https://doi.org/10.1109/68.275494
7. W. Shin, B.-A. Yu, Y.L. Lee, T.J. Yu, T.J. Eom, Y.-C. Noh, J. Lee, D.-K. Ko, Tunable Q-switched erbium-doped fiber laser based on digital micro-mirror array. Opt. Exp. **14**, 5356–5364 (2006). https://doi.org/10.1364/OE.14.005356
8. N.H. Holstein-Rathlou, K.P. Yip, O.V. Sosnovtseva, E. Mosekilde, Synchronization phenomena in nephron-nephron interaction. Chaos **11**, 417–426 (2001). https://doi.org/10.1063/1.1376398
9. E. Mosekilde, Y. Maistrenko, D. Postnov, *Chaotic synchronization: applications to living systems* (World Scientific, Singapore, 2002)
10. A.S. Pikovsky, M. Rosenblum, J. Kurths, *Synchronization: a universal concept in nonlinear sciences* (Cambridge University Press, Cambridge, 2003)
11. L. Kocarev, K.S. Halle, K. Eckert, L.O. Chua, U. Parlitz, Experimental demonstration of secure communications via chaotic synchronization. Int. J. Bifurcat. Chaos **2**(3), 709–713 (1992). https://doi.org/10.1142/S0218127492000823
12. K.M. Cuomo, A.V. Oppenheim, S.H. Strogatz, Synchronization of Lorenz-based chaotic circuits with applications to communications. IEEE Trans. Circuits Syst. II Analog Digit. Signal **40**(10), 626–633 (1993). https://doi.org/10.1109/82.246163
13. S. Jafari, M. Haeri, M.S. Tavazoei, Experimental study of a chaos-based communication system in the presence of un-known transmission delay. Int. J. Circuit Theor. Appl. **38**, 1013–1025 (2010). https://doi.org/10.1002/cta.607
14. C.J. Luo, *Dynamical system synchronization* (Springer, New York, 2013)
15. L.M. Pecora, T.L. Carroll, Synchronization in chaotic systems. Phys. Rev. Lett. **64**, 521–524 (1990). https://doi.org/10.1103/PhysRevLett.64.821
16. H. Fujisaka, T. Yamada, Stability theory of synchronized motion in coupled-oscillator systems. Prog. Theor. Phys. **69**, 32–47 (1983). https://doi.org/10.1103/PhysRevE.50.1874
17. A. Maritan, J. Banavar, Chaos noise and synchronization. Phys. Rev. Lett. **72**, 1451–1454 (1994). https://doi.org/10.1103/PhysRevLett.72.1451
18. G.I. Dykman, P.S. Landa, Y.I. Neymark, Synchronizing the chaotic oscillations by external force. Chaos Solit. Fract. **1**, 339–353 (1991). https://doi.org/10.1016/0960-0779(91)90025-5
19. M.G. Rosenblum, A.S. Pikovsky, J. Kurths, From phase to lag synchronization in coupled chaotic oscillators. Phys. Rev. Lett. **78**, 4193–4196 (1997). https://doi.org/10.1103/PhysRevLett.78.4193

20. N.F. Rulkov, M.M. Sushchik, L.S. Tsimring, H.D.I. Abarbanel, Generalized synchronization of chaos in directionally coupled chaotic systems. Phys. Rev. E **51**, 980–994 (1995). https://doi.org/10.1103/PhysRevE.51.980
21. V. Astakhov, M. Hasler, T. Kapitaniak, A. Shabunin, V. Anishchenko, Effect of parameter mismatch on the mechanisms of chaos synchronization loss in coupled systems. Phys. Rev. E **58**, 5620–5628 (1998). https://doi.org/10.1103/PhysRevE.58.5620
22. H.U. Voss, Anticipating chaotic synchronization. Phys. Rev. E **61**, 5115–5119 (2000). https://doi.org/10.1103/PhysRevE.61.5115
23. R.J. Reategui, Dynamic of complex systems with parametric modulation: duffing oscillators and a fiber laser. Doctoral dissertation, PhD Thesis, Centro de Investigaciones en Óptica, León, Gto, México (2004)
24. A. Buscarino, L. Fortuna, R. Stoop, *Advances on nonlinear dynamics of electronic systems*, vol. 17 (World Scientific, Singapore, 2019), pp. 91–96

Chapter 2
Anti-synchronization in a Pair of Coupled Multistable Systems

A. Ruiz-Silva, B. B. Cassal-Quiroga, and H. E. Gilardi-Velázquez

Abstract The main objective of this chapter is to investigate the anti-synchronization phenomenon in a pair of coupled multistable systems, generated with a piecewise-linear (PWL) system based on the Jerk equation. Anti-synchronization is an important type of synchronization for a pair of systems coupled in a master–slave scheme; systems' anti-synchronization is said to occur when coupled systems have the same absolute values but opposite signs. Using Lyapunov's stability theory, the emergence of a stable collective behavior is investigated considering different internal couplings (in one, two, or three state variables), in addition to a negative coupling. The results show that the variations in internal coupling were the result of the type of anti-synchronization: complete or partial. Finally, the effectiveness of the results obtained with some numerical simulations is verified.

Keywords Multistable systems · Anti-synchronization · Master–slave configuration

2.1 Introduction

The emergence of ordered patterns is a fundamental aspect for the proper functioning of physical, chemical, biological, and social systems, where it is important to note that without adequate coordination of the states and interconnections of each of the components, the functioning of a whole as one collective entity is impossible [1]. Various collective behaviors have been identified, such as identical synchronization,

A. Ruiz-Silva (✉)
Departamento de Matemáticas, Universidad de Sonora, Hermosillo, Sonora, México
e-mail: adriana.ruizsilva@unison.mx

B. B. Cassal-Quiroga · H. E. Gilardi-Velázquez
Facultad de Ingeniería, Universidad Panamericana, Aguascalientes, Aguascalientes, México
e-mail: hgilardi@up.edu.mx

G. Huerta Cuellar et al. (eds.), *Complex Systems and Their Applications*,
https://doi.org/10.1007/978-3-031-02472-6_2

phase synchronization, generalized synchronization, cluster synchronization, anti-synchronization, and so forth (see [1–5] and the references therein).

Recently, the mentioned collective behaviors have been extensively studied in monostable systems [3, 4]. However, there are few research works focused on the problems of emergence of synchronous behaviors for a pair of multistable systems. In this sense, the most familiar collective behavior is the identical synchronization of multistable systems, where the synchronization phenomenon can be defined as the process where two or more dynamical systems adjust their movement to common behavior as time tends to infinity [6–10].

On the other hand, for a pair of coupled multistable systems, it is also possible to study other types of collective behaviors such as anti-synchronization [7, 11, 12]. In the master–slave scheme, this behavior is characterized by the fact that the slave system has the same amplitude but opposite signs as those of the master system. Therefore, the sum of the state variables decays to zero asymptotically when the anti-synchronization appears [12, 13]. Studies of this collective behavior have been important in the sense that they constitute a relevant basis for secure communications, where the security and secrecy of the messages that are communicated can be strengthened, through the alternation of different collective behaviors. For example, alternating between an identical synchronization and an anti-synchronization in the process of digital signal transmission [13–15].

Motivated by the above discussion, we have decided to take a pair of identical multistable systems, and we study the anti-synchronization of these systems coupled in a master–slave configuration. Then, the goal of anti-synchronization problem is to use the output of the master system to control the slave system. In order that the errors decay to zero asymptotically, that is to say, the sum of the state variables is equal to zero.

In our work, we first gave necessary and sufficient conditions for the existence of the anti-synchronization solution for a pair of one class of multistable systems, which is generated with a piecewise-linear (PWL) system based on the Jerk equation and proposed by Gilardi et al. [16]. Next, we study the different effects on the anti-synchronization solution under changes in the nature of the inner coupling scheme, and variations in the coupling strength. To achieve our objectives, it is considered that the master–slave system is coupled through one, two, or three variables. Hence, two sets of internal coupling matrices are derived to achieve different anti-synchronized behaviors, such as complete or partial anti-synchronization. Furthermore, all criteria are constructed using Lyapunov functions. Finally, the results are validated using numerical simulations.

2.2 System Model

In this section, we first give some definitions and results. Consider the following multistable system, which is described by a set of three coupled differential equations that make use of the $round(x)$ function for switching among different

phase-space regions, such that the system displays infinite attractors along one dimension [16]. Therefore, the state equations of multistable systems are given by

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 \\ \dot{x}_3 &= -a_{31}x_1 - a_{32}x_2 - a_{33}x_3 + b_3(x), \end{aligned} \tag{2.1}$$

where x_1, x_2, x_3 are the three state variables, a_{31}, a_{32}, and a_{33} are the set of parameters that define the behavior of the dynamic, and nonlinear function $b_3(x)$ is defined by

$$b_3(x) = c * round\left(\frac{x_1}{\alpha}\right). \tag{2.2}$$

The above equation is considered as the commutation law of (2.1), where $c \in \mathbb{R}$ corresponds to the amplitude, and $\alpha \in \mathbb{R}$ corresponds to the length of the step given by the round function centered in the origin. It is important to mention that if α corresponds to the distance between two continuous equilibrium points given a value of c, then each equilibrium point is located exactly at the middle of two consecutive commutation surfaces [9, 16].

For simplicity, we can write (2.1) in compact form as

$$\dot{x} = Ax + B(x), \tag{2.3}$$

where $A \in \mathbb{R}^{3\times 3}$ is a constant matrix, and $B = (0, 0, b_3(x))^\top \in \mathbb{R}^3$ is the affine vector.

It should be noted that the behavior of the system (2.3) is determined by the eigenvalues of the matrix A, which can generate a wide variety of combinations and, therefore, various dynamic behaviors. In particular, Gilardi et al. [16] proposed a parameter set: $a_{31} = 14.91$, $a_{32} = 9.94$, $a_{33} = 0.994$, $c = 6.3$, and $\alpha = 0.6$ to generate a multistable system along the x_1 axis.

In order to investigate the anti-synchronization phenomenon in two coupled multistable systems of the form (2.3), a master–slave configuration with various types of couplings is investigated.

2.2.1 *Types of Coupling and Some Definition*

There are a variety of ways in which the system mentioned can be coupled. In this sense, we consider a pair of identical multistable systems defined by Eq. (2.3) coupled in a master–slave configuration as follows:

$$\dot{x} = Ax + B(x) \tag{2.4}$$

$$\dot{y} = Ay + B(y) - g\Gamma(y - qx), \tag{2.5}$$

where $x = [x_1, x_2, x_3]^\top$ and $y = [y_1, y_2, y_3]^\top$ are the state vectors of the master and slave system, respectively. $A \in \mathbb{R}^{3\times3}$, $B \in \mathbb{R}^3$ are as described before. The constant $g > 0$ is the coupling strength between the systems, and $\Gamma = Diag(\gamma_1, \gamma_2, \gamma_3) \in \mathbb{R}^{3\times3}$ is a zero-one diagonal matrix that describes the internal coupling between the multistable systems, which is constructed as follows: if $\gamma_i = 1$ indicates the master and slave systems are coupled through their i-th state variable, and $\gamma_i = 0$ otherwise. Additionally, the parameter $q \in \{-1, 1\}$ defines the type of coupling, that is, $q = 1$ is an attractive coupling, while $q = -1$ is a repulsive coupling.

There are several definitions for a collective behavior between two coupled systems, such as identical synchronization, generalized synchronization, cluster synchronization, anti-synchronization, among others [3, 4]. In particular, the anti-synchronization constitutes an important type of collective behavior, which is a classic characteristic of nonlinear dynamical systems that collaborate through repulsive coupling [11–13]. In a nutshell, anti-synchronization is a phenomenon in which the state vectors of slave systems have the same absolute amplitude but opposites signs as those of the master–slave. Consequently, we adopt the following definition:

Definition 1 A pair of dynamical systems (2.4)–(2.5) is said to achieve complete anti-synchronization (CAS), if

$$\lim_{t\to\infty} \|x_i + y_i\| = 0, \text{ for } i = 1, 2, 3, \tag{2.6}$$

where the symbol $\|\cdot\|$ denotes the Euclidean norm of a vector [5, 17].

However, when the condition (Eq. (2.6)) is not satisfied for all states, that is, at least one pair of variables maintains a constant distance that emerges as a conserved quantity of which is sensitive to changes in the initial states. In these cases, the collective behavior is known as partial anti-synchronization, and mathematically, it is defined as:

Definition 2 A pair of dynamical systems (2.4)–(2.5) is said to achieve partial anti-synchronization (PAS), if at least one of the states

$$\lim_{t\to\infty} \|x_k + y_k\| = c_k, \text{ for some } k \in \{1, 2, 3\}, \tag{2.7}$$

where c_k is a constant, and the rest of the states satisfies Eq. (2.6) [7, 13].

In order to establish necessary and sufficient conditions to guarantee the anti-synchronization of the master–slave system, the error system and its stability must be analyzed around its zero solution.

2.3 Theoretical Analysis

We define the anti-synchronization error vector between the master–slave system as the sum of the state variables, i.e., $E = [x + y]^T = [e_1, e_2, e_3]^T \in \mathbb{R}^3$, where $e_i = x_i + y_i \in \mathbb{R}$ for $i = 1, 2, 3$, is each variable's state error. So that the error dynamics is given as follows:

$$\dot{E} = A(x + y) + B(x) + B(y) - g\Gamma(y + qx). \tag{2.8}$$

Next, let q is a control parameter that influences the dynamics of the coupling between the master and the slave. Substituting $q = -1$ in Eq. (2.8), the error dynamics simplifies to

$$\dot{E} = (A - g\Gamma)E + B(x) + B(y). \tag{2.9}$$

In an explicit form, we have

$$\begin{bmatrix} \dot{e}_1 \\ \dot{e}_2 \\ \dot{e}_3 \end{bmatrix} = \begin{bmatrix} -g\gamma_1 e_1 + e_2 \\ -g\gamma_2 e_2 + e_3 \\ -a_{31}e_1 - a_{32}e_2 - (a_{33} + g\gamma_3)e_3 + (b_3(x) + b_3(y)), \end{bmatrix} \tag{2.10}$$

where $b_3(\cdot)$ is the third entry of the affine vector, B, defined in (2.2).

Note that the internal coupling matrix and the coupling strength play a crucial role in determining whether the origin is an equilibrium point of the error system (2.10), and consequently, whether complete or partial anti-synchronization is achieved between the master–slave systems.

To demonstrate the anti-synchronization arises between the master–slave system is to show that the error dynamics has an equilibrium point and has a fixed point, and by constructing a Lyapunov function, it can be shown that the error system is globally stable. Our results are presented in the following proposition:

Proposition 1 *Consider a master–slave system described by (2.4)–(2.5) with* $\Gamma \in \{Diag(1, 1, 1), Diag(1, 0, 1), Diag(1, 1, 0), Diag(1, 0, 0)\}$. *If there are symmetric positive definite matrix* $Q \in \mathbb{R}^{3\times3}$, *and a sufficient large positive constant* $d^* > 0$, *such that*

$$(A - d^*\Gamma) + (A - d^*\Gamma)^T \leq -Q, \tag{2.11}$$

with $g \geq d^*$. *Consequently, the master–slave system (2.4)–(2.5) presents complete anti-synchronization.*

Proof Suppose that matrix $\Gamma = Diag(\gamma_1, \gamma_2, \gamma_3) \in \mathbb{R}^{3\times3}$ can be take different configurations as $\{Diag(1, 1, 1), Diag(1, 0, 1), Diag(1, 1, 0), Diag(1, 0, 0)\}$.

So, solving for the equilibrium point, we find

$$\begin{aligned} e_1^* &= \frac{b_3(x) + b_3(y)}{a_{31} + a_{32}g\gamma_1 + (a_{33} + g\gamma_3)g^2\gamma_1\gamma_2} \\ e_2^* &= g\gamma_1 e_1^*,\ e_3^* = g^2\gamma_1\gamma_2 e_1^*. \end{aligned} \tag{2.12}$$

As $b_3(x) + b_3(y)$ is bounded, e_1^* tends to zero when g tends to infinity. Also, if e_1^* tends to zero, then e_2^* and e_3^* also tend to zero. Therefore, the error system has $(0, 0, 0)^T$ as its sole equilibrium point.

Second, we consider the candidate Lyapunov function $V = E^T E$, and using a similar procedure as in [18, 19]. So, the time derivative of V along the trajectories of the error dynamics (2.9) is given by

$$\dot{V} = E^T\Big[(A - g\Gamma) + (A - g\Gamma)^T\Big]E + E^T\Big[B(x) + B(y)\Big]. \tag{2.13}$$

Let $g = d^*$, and using (2.11), we obtain

$$\dot{V} = -E^T Q^T E + E^T\Big[B(x) + B(y)\Big]. \tag{2.14}$$

Assuming that around the zero solution, the term $B(x) + B(y)$ can be seen as fading perturbation that satisfies $\|B(x) + B(y)\| \leq \beta\|x + y\|$ with $\beta \leq 0.5\lambda_{min}(Q)$, where λ_{min} denotes the minimum eigenvalue of the matrix Q. Therefore,

$$\dot{V} = E^T(-Q + \beta I_3)E < 0, \tag{2.15}$$

where I_3 is an identity matrix of size 3×3.

Finally, according to the previous arguments, this establishes that the origin is stable. As a consequence, the master–slave system achieves the complete anti-synchronization. □

The previous proposition reduces the anti-synchronization problem to find a coupling strength value, g, such that the matrix $A - g\Gamma$ only has eigenvalues with a negative real part. Moreover, due to the linear operator A satisfies the definition of unstable dissipative systems type I [20, 21], the coupling strength between the master–slave system must be greater than the positive real part of the complex eigenvalues that guarantees that $A - g\Gamma$ is Hurwitz. However, changes in the initial conditions or even a larger coupling strength can change the resulting coupling solution.

On the other hand, it is possible to achieve partial anti-synchronization considering that the internal coupling matrix Γ belongs to the set $\{Diag(0, 1, 0), Diag(0, 0, 1), Diag(0, 1, 1)\}$. Since in these cases, the equilibrium point of the error system has the form $(e_1^*, 0, 0)^T$, where $e_1^* = c_1$ is a constant depending on the initial conditions of the states of the master–slave systems. However, it is possible

to choose a Lyapunov function corresponding to the dynamics of the error and show that the equilibrium point is stable. Therefore, we present the following proposition:

Proposition 2 *Consider a master–slave system described by (2.4)–(2.5), with* Γ *belongs to the set* $\{Diag(0,1,0),\ Diag(0,0,1),\ Diag(0,1,1)\}$. *There the error system (2.9) is stable about* $(e_1^*,0,0)^T$, *if the coupling strength satisfies*

$$g \geq d^* \tag{2.16}$$

with c_1 *is a constant depending on the initial conditions of the states, and* d^* *is a critical value that holds (2.11). As a consequence, the master–slave system achieves the partial anti-synchronization in the sense of Definition 2.*

Proof Considering the Eq. (2.10), it is easy to see that the equilibrium point is given by $e_1^* = (b_3(x) + b_3(y))/a_{31}$, $e_2^* = 0$, and $e_3^* = 0$. Using the argument similar to $(b_3(y) + b_3(x))$ is bounded, and thus e_1^* =constant= c_1 is a constant that depends on the initial conditions of the states. Consequently, the anti-synchronization error system has $E^* = (e_1^*, 0, 0)^T$ as its equilibrium point.

Let us choose a Lyapunov function $V = (E - E^*)^T (E - E^*)$, and using a similar procedure to the previous proposition. The time derivative of the Lyapunov function along the trajectory of the equilibrium point is given by

$$\dot{V} = 2\sum_{i=1}^{3}(e_i - e_i^*)\dot{e}_i. \tag{2.17}$$

Thus, if $(A - d^*\Gamma) + (A - d^*\Gamma)^T \leq 0$, which satisfies when $g \geq d^*$, we obtain that $\dot{V} \leq 0$. Consequently, the master–slave system achieves the partial anti-synchronization. □

In this section, sufficient conditions were determined to achieve anti-synchronization of a pair of systems coupling in a master–slave configuration. In order to illustrate these dynamic behaviors in the next section, we present some numerical examples.

2.4 Numerical Results

The effectiveness of the proposed scheme is shown in this section. Numerical simulations are done by using MATLAB.

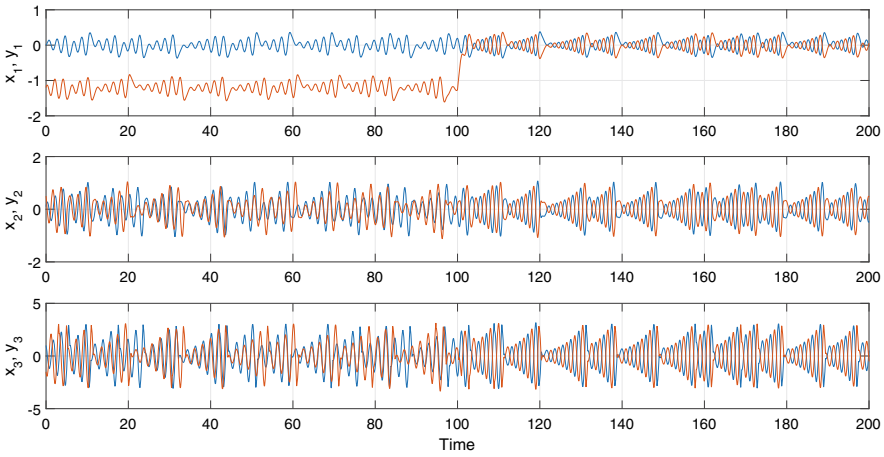


Fig. 2.1 Numerical simulation for master–slave system (2.4)–(2.5) with $\Gamma = Diag(1, 1, 1)$, and the coupling strength $g = 1$ applied at 100 time units. The time is given by arbitrary units (a.u.). Color code: Blue line corresponds to master system, and the orange line to slave system

2.4.1 Complete Anti-synchronization

Example 1 Let the master–slave system (2.4)–(2.5), with the common linear operator A, and the vector B defined in Sect. 2.2. Additionally, we consider that the internal coupling matrix is $\Gamma = Diag(1, 1, 1)$, and the initial conditions are given by $x_0 = [0,\ 0.1,\ 1]^T$ and $y_0 = [-1.2,\ 0.5,\ -1.5]^T$.

According to the theory discussed in the previous section, using the Proposition 1, we obtain that if $g > 0.42$, thus the matrix $A - g\Gamma$ will have only eigenvalues with negative real part, which should lead the master–slave system to achieve the complete anti-synchronization, and the results of the numerical simulations are shown in Figs. 2.1 and 2.2.

Figure 2.1 shows the time series of coupled systems; the blue line corresponds to master system, while the orange line represents the solution of slave system. It is considered that for $t < 100$ the systems are uncoupled, so each solution evolves in its own attractor. After 100 time units, the master–slave system is connected with a coupling strength of $g = 1$, and it is observed how the slave system solution converges to the attractor located at the origin but oscillates in anti-phase in relation to the master system. Additionally, Fig. 2.2 shows the numerical verification where the complete anti-synchronization is achieved between the three states of the coupled systems. In this figure, it is possible to appreciate how all the errors converge to zero.

Example 2 Now, we consider the same master–slave system, but the internal coupling matrix is $\Gamma = Diag(1, 0, 1)$, and the initial conditions are given by $x_0 = [0.6,\ 0.1,\ 1.5]^T$ and $y_0 = [-4.2,\ 0.5,\ -1]^T$. In this case, using the

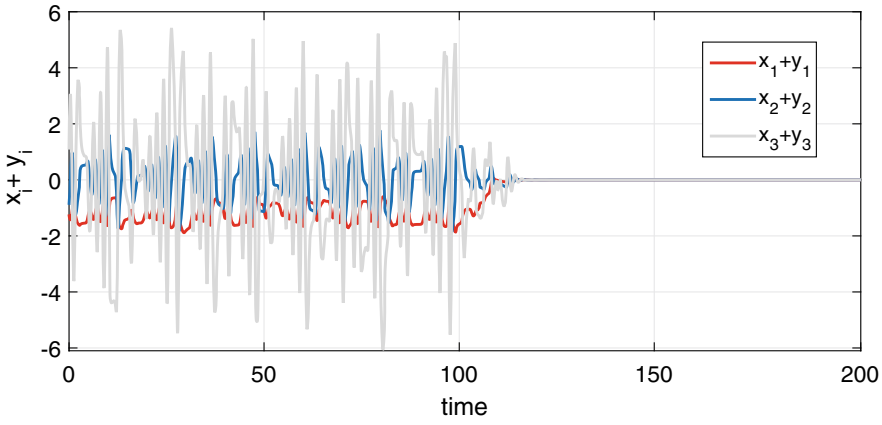


Fig. 2.2 Error evolution for master–slave system (2.4)–(2.5) with $\Gamma = Diag(1, 1, 1)$, and the coupling strength $g = 1$ applied at 100 time units. The time is given by arbitrary units (a.u.). Color code: The red line represents the error e_1, blue line the error e_2, and the gray line the error e_3

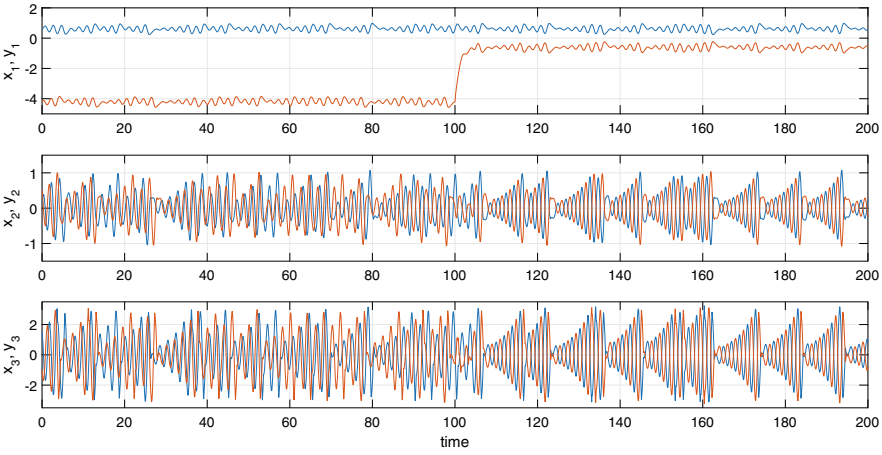


Fig. 2.3 Numerical simulation for the master–slave system (2.4)–(2.5) using the internal coupling matrix $\Gamma = Diag(1, 0, 1)$, and the coupling strength $g = 1$ applied at 100 time units. The time is given by arbitrary units (a.u.). Color code: Blue line corresponds to master system, and the orange line to slave system

Proposition 1, we obtain the critical value of the coupling strength under which anti-synchronization is achieved. Particularly for this example, the minimum coupling strength is $g \geq 0.8$. Therefore, for the numerical simulation, it is considered that the coupling strength $g = 1$, and the results are presented in Figs. 2.3 and 2.4.

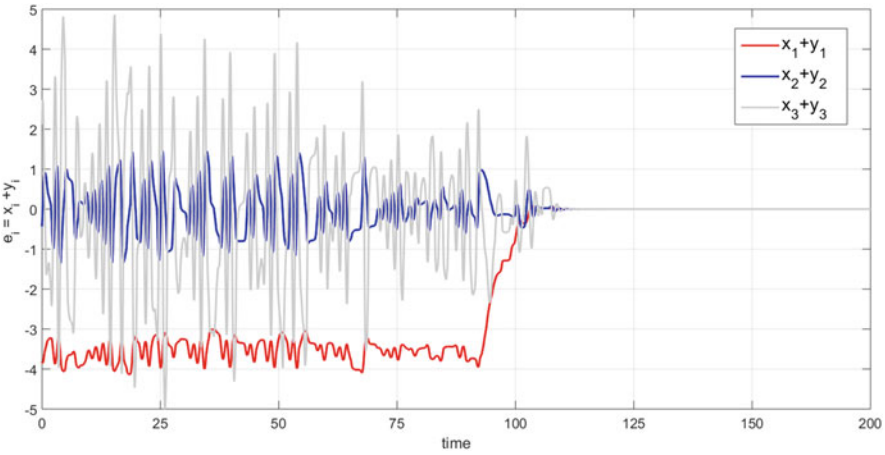


Fig. 2.4 Error evolution for master–slave system (2.4)–(2.5) with $\Gamma = Diag(1, 0, 1)$, and the coupling strength $g = 1$ applied at 100 time units. The time is given by arbitrary units (a.u.). Color code: The red line represents the error e_1, blue line the error e_2, and the gray line the error e_3

In Fig. 2.3, it is considered that the blue solution belongs to the master system, and the orange line corresponds to the slave system. Similar to the previous example, initially the systems are uncoupled, and we can observe that systems oscillate in its corresponding attractor. After 100 time units, the master–slave systems are coupled, and it is possible to observe that the slave system solutions converge to the attractor generated for the initial condition $-x_0$. Moreover, it can be seen that the trajectories of the coupled systems evolve in anti-phase, but in different attractors. Furthermore, if the sum between the state variables is calculated, then there is the numerical verification that the error converges to zero in the three states, which is presented in Fig. 2.4.

It is important to mention that these examples can be interesting because it illustrates a difference between a collective anti-phase behavior and the anti-synchronization phenomenon. In the first example, since the initial condition of the master system is around the origin, it is not possible to appreciate the movement of the slave solution, and the resulting behavior can be confused with an anti-phase synchronization. Instead, in Example 2, it is possible to observe how the slave system solution changes the resulting attractor, since it converges to the attractor generated by the initial conditions, $-x_0$, and not to those of the master system. But the oscillation between the systems remains in anti-phase, and the three states to the error system converge to zero.

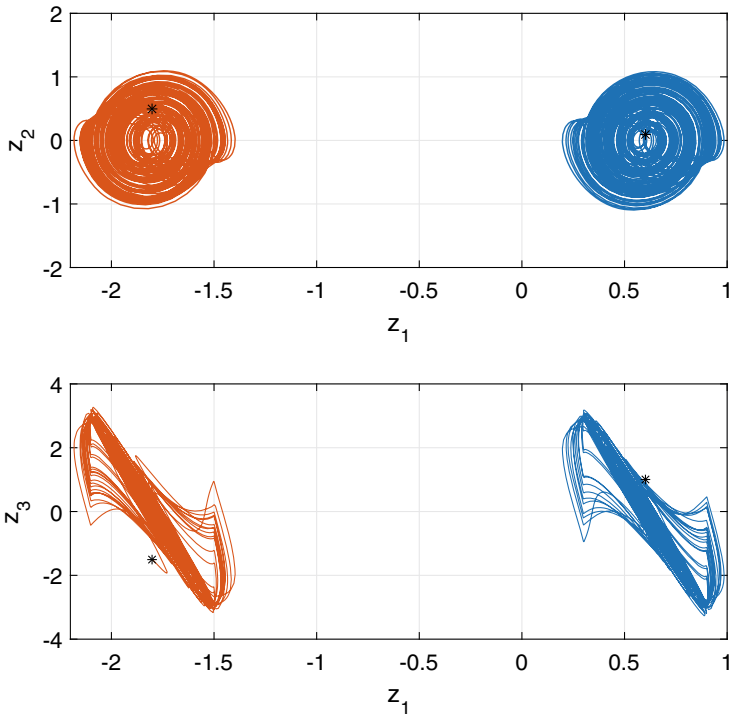


Fig. 2.5 Projections of trajectory solutions onto (**a**) the (z_1, z_2) plane and (**b**) (z_1, z_3) plane with $z = x, y$, for the master–slave systems (2.4)–(2.5) with $\Gamma = Diag(0, 1, 0)$ and coupling strength $g = 1$. The black asterisks indicate the initial conditions

2.4.2 *Partial Anti-synchronization*

Next, we consider the same master–slave configuration, but now using the internal coupling matrix belongs to $\{Diag(0, 1, 0),\ Diag(0, 0, 1),\ Diag(0, 1, 1)\}$. Thus, with this configuration, it is possible to achieve the partial anti-synchronization.

Example 3 Consider master–slave system (2.4)–(2.5) with $\Gamma = Diag(0, 1, 0)$, and the initial conditions $x_0 = [0.6, .1, 1]^T$ and $y_0 = [-1.8, 0.5, -1.5]^T$. Using the Proposition 2 can be determined to achieve the partial anti-synchronization in the structure presented. In this case, the minimum coupling strength is $g_{min} = 0.88$; for simplicity, we consider that $g = 1$.

Figure 2.5 shows the projections of solutions onto the planes (a) (z_1, z_2) and (b) (z_1, z_3) for each of the systems coupled where $z = x, y$. The blue line corresponds to the master systems, orange line represents the slave system, and the set of initial conditions is marked with an asterisk. For this case, when analyzing the projections of both systems, apparently, the coupling does not affect the slave system.

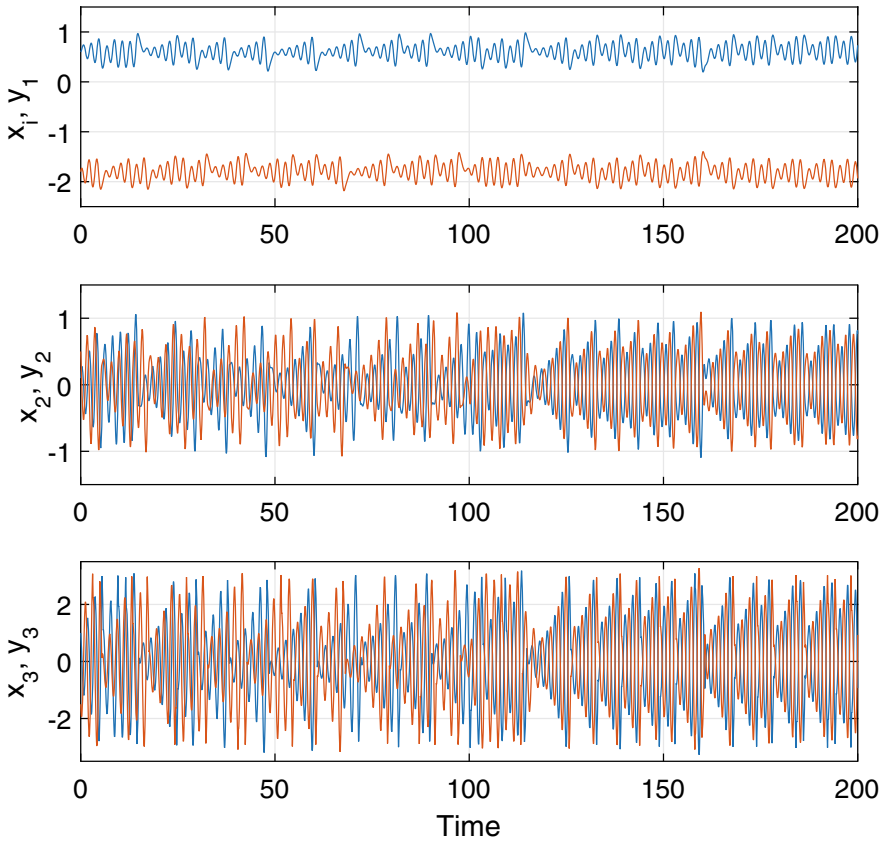


Fig. 2.6 Numerical simulation for the master–slave system (2.4)–(2.5) using the internal coupling matrix $\Gamma = Diag(0, 1, 0)$, and the coupling strength $g = 1$ applied at 100 time units. So, the blue line corresponds to the master system, and the orange line corresponds to the slave system. The time is given by arbitrary units (a.u.)

On the other hand, Fig. 2.6 shows the evolution of the solutions in the three states, where after 100 time units the coupling between both systems is applied, and it is possible to observe that there is an evolution in anti-phase for the three states. However, in this example, the slave system does not tend to the attractor located in the mirror of the master; rather, it remains in its original attractor. Therefore, when the errors are analyzed, it is possible to observe how for the first state the error tends to $c_1 = -1.2$ (see Fig. 2.7), whereas the other states of the errors tend to zero. In this sense, the master–slave system achieves partial anti-synchronization.

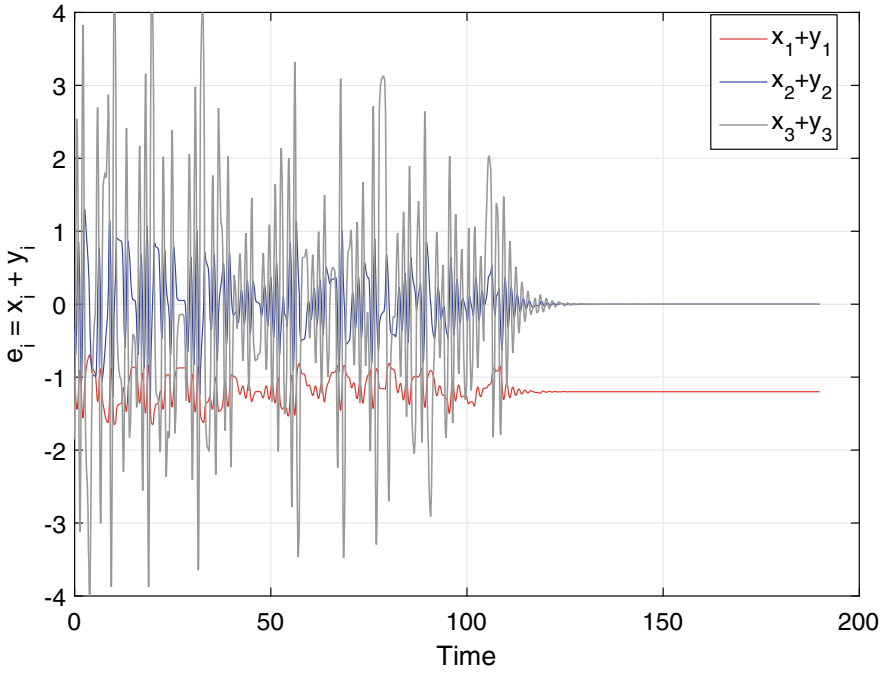


Fig. 2.7 Error evolution for master–slave system (2.4)–(2.5) with $\Gamma = Diag(0, 1, 0)$, and the coupling strength $g = 1$ applied at 100 time units. The red line represents the error e_1, blue line the error e_2, and the gray line the error e_3. The time is given by arbitrary units (a.u.)

2.5 Conclusion

In this chapter, we focused on studying and observing the anti-synchronization for a pair of identical multistable systems coupled in a master–slave configuration, using the Lyapunov stability analysis. In general, the anti-synchronization of the master–slave system was observed, where the nature of the internal coupling matrix Γ has a great influence in determining the type of anti-synchronization, such as: complete and partial anti-synchronization, and the coupling strength is important to achieve stable collective behavior.

For first instance, our study was focused on the complete anti-synchronization, and our numerical results show that for sufficiently large coupling strength the solution of the slave system tends to the attractor generated in the basin around $-x_0$, but it presents anti-phase oscillations for all states. In general, the solutions satisfy the conditions of being oscillations of the same amplitude, but of opposite signs.

For second instance, we consider a master–slave system, but the internal coupling matrix does not consider x coupling, and it was observed that the resulting collective behavior is a partial anti-synchronization, where the error for the first states is a

constant that depends on the initial conditions of the states of the coupled systems. It should be mentioned that in this scenario, the slave system does not change the basin of attraction once it has been connected with the master system, and the only observable difference is the phase shift of the oscillations of the slave system. In this sense, this type of synchronous behavior could also be considered as an anti-phase synchronization [3].

Finally, we strongly believe that the methodology discussed here can be applied to a wide class of PWL systems with chaotic behavior, such as those presented in [20, 22]. On the other hand, the results can be directly extended for a network of N identical multistable systems with linear coupling functions and directed topology of the network. Here we explore two scenarios: the anti-synchronization for all nodes in the network, or mixed synchronization between the nodes of the networks.

References

1. S.C. Manrubia, A.S. Mikhailov, D. Zanette, *Emergence of Dynamical Order: Synchronization Phenomena in Complex Systems*, vol. 2 (World Scientific, Singapore, 2004)
2. L.M. Pecora, T.L. Carroll, G.A. Johnson, D.J. Mar, J.F. Heagy, Fundamentals of synchronization in chaotic systems, concepts, and applications. Chaos Interdiscip. J. Nonlinear Sci. **7**(4), 520–543 (1997)
3. S. Boccaletti, J. Kurths, G. Osipov, D.L. Valladares, C.S. Zhou, The synchronization of chaotic systems. Phys. Rep. **366**(1–2), 1–101 (2002)
4. L.M. Pecora, T.L. Carroll, Synchronization of chaotic systems. Chaos Interdiscip. J. Nonlinear Sci. **25**(9), 097611 (2015)
5. L. Ren, R. Guo, U.E. Vincent, Coexistence of synchronization and anti-synchronization in chaotic systems. Arch. Control Sci. **26**(1), 69–79 (2016)
6. E. Jiménez-López, J.S. González Salas, L.J. Ontañón-García, E. Campos-Cantón, A.N. Pisarchik, Generalized multistable structure via chaotic synchronization and preservation of scrolls. J. Franklin Inst. **350**(10), 2853–2866 (2013)
7. C. Hens, S.K. Dana, U. Feudel, Extreme multistability: Attractor manipulation and robustness. Chaos Interdiscip. J. Nonlinear Sci. **25**(5), 053112 (2015)
8. A. Anzo-Hernández, E. Campos-Cantón, M. Nicol, Itinerary synchronization between PWL systems coupled with unidirectional links. Commun. Nonlinear Sci. Numer. Simul. **70**, 102–124 (2019)
9. A. Ruiz-Silva, H.E. Gilardi-Velázquez, E. Campos, Emergence of synchronous behavior in a network with chaotic multistable systems. Chaos Solitons Fractals **151**, 111263 (2021)
10. J.L. Echenausía-Monroy, L.J. Ontañón-García, J. Pena Ramirez, On synchronization of unidirectionally coupled multi-scroll systems: Dynamic vs static interconnections**partially supported by Conacyt Mexico under grant a1-s-26123. IFAC-PapersOnLine **54**(17), 53–58 (2021). 6th IFAC Conference on Analysis and Control of Chaotic Systems CHAOS 2021
11. P. Chakraborty, S. Poria, Extreme multistable synchronisation in coupled dynamical systems. Pramana **93**(2), 1–13 (2019)
12. V. Kamdoum Tamba, V.T. Pham, D. Vo Hoang, S. Jafari, F.E. Alsaadi, F.E. Alsaadi, Dynamic system with no equilibrium and its chaos anti-synchronization. Automatika **59**(1), 35–42 (2018)
13. R. Guo, Y. Qi, Partial anti-synchronization in a class of chaotic and hyper-chaotic systems. IEEE Access **9**, 46303–46312 (2021)

14. T. Yang, L.O. Chua, Secure communication via chaotic parameter modulation. IEEE Trans. Circuits Syst. I Fundam. Theory Appl. **43**(9), 817–819 (1996)
15. A.D. Pano-Azucena, J. de Jesus Rangel-Magdaleno, E. Tlelo-Cuautle, A. de Jesus Quintas-Valles, Arduino-based chaotic secure communication system using multi-directional multi-scroll chaotic oscillators. Nonlinear Dyn. **87**(4), 2203–2217 (2017)
16. H.E. Gilardi-Velázquez, L.J. Ontañón-García, D.G. Hurtado-Rodriguez, E. Campos-Cantón, Multistability in piecewise linear systems versus eigenspectra variation and round function. Int. J. Bifurc. Chaos **27**(09), 1730031 (2017)
17. J. Hu, S. Chen, L. Chen, Adaptive control for anti-synchronization of Chua's chaotic system. Phys. Lett. A **339**(6), 455–460 (2005)
18. H.K. Khalil, *Nonlinear Systems*, vol. 3 (Prentice Hall, 2002)
19. J. Manuel M.S. Gonçalves, Constructive global analysis of hybrid systems. PhD thesis, Massachusetts Institute of Technology, 2000
20. A. Anzo-Hernández, H.E. Gilardi-Velázquez, E. Campos-Cantón, On multistability behavior of unstable dissipative systems. Chaos Interdiscip. J. Nonlinear Sci. **28**(3), 033613 (2018)
21. J.L. Echenausía-Monroy, G. Huerta-Cuéllar, A novel approach to generate attractors with a high number of scrolls. Nonlinear Anal. Hybrid Syst. **35**, 100822 (2020)
22. J.L. Echenausía-Monroy, J.H. García-López, R. Jaimes-Reátegui, D. López-Mancilla, G. Huerta-Cuéllar, Family of bistable attractors contained in an unstable dissipative switching system associated to a SNLF. Complexity **2018**, 9 (2018)

Chapter 3
Analysis of Synchronizability in Small-World Complex Networks

E. Zambrano-Serrano, C. Posadas-Castillo, M. A. Platas-Garza, and J. R. Rodríguez-Cruz

Abstract The research related to complex networks is an active field mainly inspired by its broad ability to model a wide variety of systems. For instance, we observe biological networks at the microscopic level, genetic regulation networks, protein networks, and neural networks. On the other hand, we find computer networks and social networks at a higher level of organization. For this reason, the study of complex networks is attractive, specifically those that present small-world characteristics and nonlinear dynamic systems at their nodes. This chapter presents a comparison of three algorithms for generating small-world complex networks, Watts–Strogatz, Newman–Watts, and Sanchez–Posadas, intending to show which algorithm enhances the synchrony of complex networks that consider chaotic oscillators in their nodes, from the perspective of *synchronizability*—the capacity of networks in synchronizing the activity of their dynamical parts. The comparison of these algorithms has been made from the eigenratio of the Laplacian matrix from the connection graph. It relates the largest eigenvalue divided by the second smallest one. The results that show the proposed by Sanchez–Posadas algorithm exhibit certain advantages over the other two studied.

Keywords Synchronizability · Small-world networks · Eigenratio · Synchronization

3.1 Introduction

Due to the fascinating applications in modeling natural and artificial systems, complex networks have been converted into a relevant research area [1]. For example, the brain is a network of neurons connected through synapses; a company is a network of people with various types of connections between them;

E. Zambrano-Serrano (✉) · C. Posadas-Castillo · M. A. Platas-Garza · J. R. Rodríguez-Cruz
Universidad Autónoma de Nuevo León, Facultad de Ingeniería Mecánica y Eléctrica, San Nicolás de los Garza, Mexico
e-mail: ernesto.zambranos@uanl.edu.mx; cornelio.posadascs@uanl.edu.mx

G. Huerta Cuellar et al. (eds.), *Complex Systems and Their Applications*,
https://doi.org/10.1007/978-3-031-02472-6_3

transportation networks, ecological networks of animals and plants, to name a few [2]. Furthermore, the collective behavior of dynamic systems over the networks has been an active research topic being the synchronization the most interesting result of collective behavior. Therefore, it has been widely investigated in different areas ranging from the flashing of fireflies, swimming of coupled pendulums, coupled chaotic systems, so on [3–5]. Moreover, these behaviors have implied different types of synchronizations such as complete, lag, phase, bubbling, and generalized synchronization, to mention a few [6].

The synchronization in complex networks is mainly related to (1) the structure network, (2) the dynamic inherent of individual systems or nodes, (3) the type of strength of the interaction among individual dynamical systems [7]. Driving two fundamental concerns: obtaining the synchronous solution and determining its stability [8]. In this scenario, an interesting question has emerged: when is such synchronous manifold stable, particularly with the coupling configuration and strength? Regarding the stability of the synchronization manifold, some necessary or sufficient conditions have been proposed [9, 10]. Based on transverse Lyapunov exponents, the master-stability-function formalism proposed by Pecora and Carroll in [9] allows the local synchronization analysis. Its significance is on the separation of the effects of network structure and individual nodes' dynamics. As a result, synchronizability, a measure to quantify the degree of synchronization, could be assessed by some topological criteria without referring to the specific node dynamics. This measure depends entirely on the structure of the connection graph. It shows that the eigenratio of the Laplacian matrix of the connection graph, i.e., the largest eigenvalue divided by the second smallest one, in which a smaller eigenratio is favored for high synchronizability [9]. In other words, the smaller is the eigenratio of a network, the better its synchronizability [7]. Some works on this subject are those described by [11–13]. In [11], the synchronizability is studied due to the time-varying coupling strength on neuronal networks formulated by the Hindmarsh–Rose model. In [12], the relationships between synchronizability and the parameters of multi-layer-coupled star-composed networks were presented. In [14], the authors show the relationship of the eigenratio parameter with the synchronization of complex networks that simulate brain connections in patients with early Alzheimer's disease. Also in [15] is presented a study that compares eigenvalue with synchronization in small-world networks.

After the pioneering work of Watts and Strogatz in small-world networks [16] pointing out the importance of network topology, much research has been devoted to proposing new algorithms that introduce the small-world properties in complex networks considering the emergence of the following characteristics: high connectivity and a short average distance between nodes [17–19]. In [19], an algorithm is introduced to generate small-world networks from regular networks by creating shortcuts to reduce the average distance between pairs of nodes. Then, the resulting network exhibits the average shortest path length and high connectivity. Furthermore, it shows that the clustering coefficient, a measure of connectivity between nodes, would be slightly affected for small changes in the topology.

It is desired to have high synchronizability for many design applications and modeling in a small-world topology. Then a question arises: given the methodologies to generate a small network (Watts–Strogatz, Newman–Watts, and Sanchez–Posadas), which present the best synchronizability value? Since the synchronizability can be improved, modifying the network structure by adding or removing edges/nodes and rewriting edges.

In this chapter, from the perspective of synchronizability, we study the synchronization phenomenon of three different algorithms' generators of small-world networks with identical systems in their nodes, where each system is a chaotic oscillator. By comparing these algorithms, it is intended to show that the algorithm proposed by Sanchez–Posadas is more efficient in generating small-world complex networks that can contribute to a better synchronization in an emergent way. The structure of this chapter is as follows: Sect. 3.2 describes the basic concepts of complex networks. In Sect. 3.3, the main characteristics of the chaotic oscillator are presented. Small-world network synchronization is presented in Sect. 3.4. Section 3.5 focuses on analyzing the results of the experiments carried out. Finally, we close the paper with the conclusions.

3.2 Complex Networks

The study of networks is a mathematics branch called graph theory. The birth of graph theory dates back to 1736 when the Swiss mathematician Leonhard Euler publicized the solution of the Königsberg bridge problem. Since then, this theory has been used in many fields of science such as biology, neurology, sociology, and economy [1]. More formally, we can define complex networks as a graph as follows:

Definition 1 A graph $G = (N, L)$ can be defined as a set of elements or nodes $(N = n_1, n_2, \ldots, n_N,)$, which have connections or links $(L = l_1, l_2, \ldots, l_k)$ that allow us to establish binary relationships between the elements of the set. Each link L is a subset of N of two elements denoting a connection among the two vertices i.e., $L \subset \{\{i, j\} : i, j \in N\}$ [2]. If two links in a node are the same, this edge is called auto-loop. The graphs can be grouped into two categories, the directed, Fig. 3.1a, and the undirected, Fig. 3.1b; a directed graph is one where the direction of interaction between nodes is limited to the direction indicated by the arrows, and in an undirected graph, the interactions between nodes are considered bidirectional [20].

The term complex refers to the characteristics of the nodes and the structure or network topology. Nodes can be modeled by differential equations or any other mathematical model. However, it can be in two main configurations: master–slave, Fig. 3.2a, and bidirectional, Fig. 3.2b. In the first model, the master imposes its dynamics on the rest of the network's nodes. Meanwhile, in the second model, due to the absence of a master, the configuration shows an emergent dynamic, which is different from that of any of the network nodes. We consider that the complex

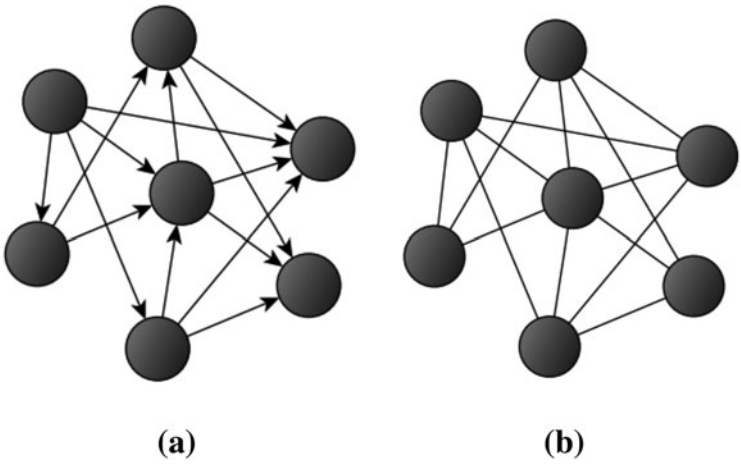


Fig. 3.1 (**a**) Direct graph; (**b**) Undirected graph

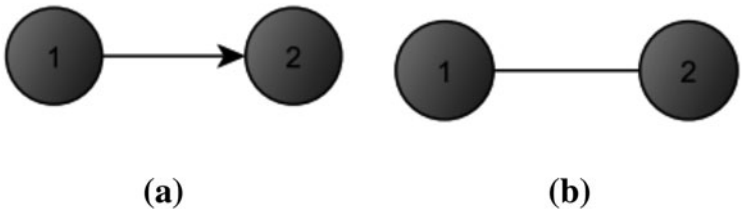


Fig. 3.2 (**a**) Master–slave configuration; (**b**) Bidirectional configuration

network has achieved a synchronization state in this scenario. This kind of behavior is what we are going to focus on.

We can organize these complex networks in different groups, structural complex networks, weighted, random, free scale, and small world:

- **Structural complex networks:** These include regular and irregular topologies; regulars shows a very well-defined behavior (ring, star, and global), but irregular complex networks do not show a well-defined behavior.
- **Weighted complex networks:** They are characterized by having weights in the links, which introduce hierarchy.
- **Random complex networks:** They are characterized by having all their connections randomly performed between pairs of nodes.
- **Free-scale complex network:** This type of network has many nodes, some with many links and others with few connections. The nodes with many links have been called hubs.
- **Small-world complex network:** This type of network is characterized by a high degree of connectivity and a small average distance between nodes due to the presence of long-range connections between distant nodes.

3.2.1 Small-World Networks

The small-world property refers to the existence of long links reach or shortcuts connecting pairs of distant nodes in the network, requiring a small number of steps to reach any node on the network. The characteristics of the complex networks affected by this property are as follows [19, 21]: the clustering coefficient, which refers to the average fraction of pairs of neighbors of a node, which are neighbors to each other. C_i is the clustering coefficient of node i and is defined as the quotient between the number E_i, which is the real existing links, and k_i, the neighbors of node i. So, we define the clustering coefficient C of the network as the average of the C_i over all nodes i as follows [22]:

$$C_i = \frac{2E_i}{k_i(k_i - 1)}, \quad 1 \leq i \leq N, \tag{3.1}$$

where N is the network size. Other important characteristic of this kind of complex network is the shortest average path L. It is defined as the shortest distance between any pairs of nodes.

$$L = \frac{1}{N(N-1)} \sum_{\forall_{i,j}, i \neq j}^{N} l_{i,j}, \quad i \leq i, \ j \leq N, \tag{3.2}$$

and l_{ij} refers to the shortest path between node i and node j. This characteristic is one of the most studied in this kind of network, due to its application in situations of real interest, such as the study of memory, capacity in neural networks, spread of epidemics, etc.

Another relevant concept is the *Laplacian matrix* or coupling matrix $A = (a_{i,j}) \in R^{n \times n}$; this concept refers to how many links exist among the network nodes. If there is a connection between nodes i and j, then the element $a_{i,j} = 1$; otherwise $a_{i,j} = 0$, with $(i \neq j)$. The diagonal elements $a_{i,i}$ of A are defined as

$$a_{i,i} = -\sum_{j=1, j \neq i}^{N} a_{i,j} = -\sum_{j=1, j \neq i}^{N} a_{j,i}; \quad \text{with } i = 1, 2, \ldots, N. \tag{3.3}$$

Thus, for graph shown in Fig. 3.3, the corresponding Laplacian matrix is

$$A = \begin{bmatrix} -3 & 1 & 1 & 1 & 0 \\ 1 & -3 & 0 & 1 & 1 \\ 1 & 0 & -3 & 1 & 1 \\ 1 & 1 & 1 & -4 & 1 \\ 0 & 1 & 1 & 1 & -3 \end{bmatrix}, \tag{3.4}$$

and its eigenvalues denoted as $\lambda_i(A) = \{0, -3, -3, -5, -5\}$, with $i = 1, \ldots, 5$.

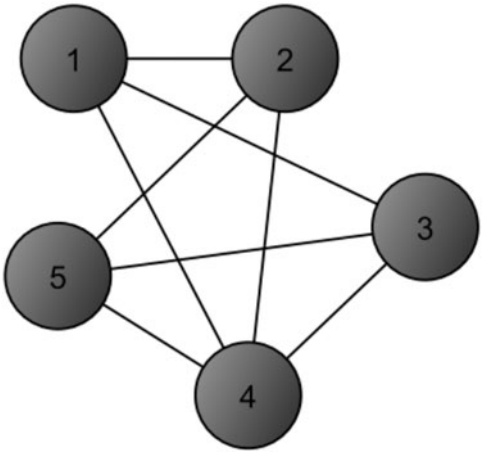


Fig. 3.3 Irregular complex network

3.2.2 Main Algorithms to Generate Small-World Complex Networks

Watts–Strogatz algorithm: In 1998, D. J. Watts and S. H. Strogatz proposed an algorithm to introduce the small-world property in a regular network. The topology of this complex network is known as the closest neighbor, which consists of a ring arrangement with periodic connection conditions. The Watts–Strogatz small-world model is created by rewiring one end of several existing links to new locations chosen randomly [16].

Newman–Watts algorithm: After Watts and Strogatz published their algorithm for generating complex small-world networks, a new version emerged the year after this pioneering work appeared. In 1999, M. E. J. Newman and D. J. Watts proposed their modified version of the original small-world algorithm. Like the previous algorithm, the Newman–Watts algorithm starts from the nearest neighbor topology. The small-world property is introduced by adding new links to randomly chosen pairs of nodes. The restrictions of this algorithm thus remain unchanged, except for the second, which is eliminated since the number of links varies as a function of probability [17].

Sanchez–Posadas algorithm: In this case, the algorithm proposed by Sanchez–Posadas is based on the addition of new connections starting from a network in a ring topology, where these new connections are created between pairs of nodes chosen randomly; in this, are they propose a method to reconnect and also create new connections [19]?

3.2.3 *Sanchez–Posadas Algorithm*

As we said previously, the algorithm starts from a network in ring topology of known size N and the periodic connection condition k as is shown in Fig. 3.4.

If probability $p = 0.1$, then $Nkp = 3$. That means we have to reconnect one of the ends of $3(Nkp)$ existing links to new positions chosen randomly and the result is displayed in Fig. 3.5.

Then, we have to create two more connections for any reconnection made before:

– A connection is created between the node that received the connection and one of the k neighbors of the node that sent the initial connection, chosen randomly. If

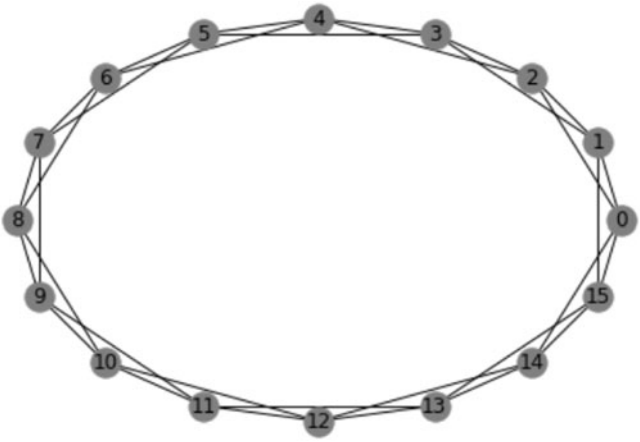


Fig. 3.4 Complex network with $N = 16$ and a connection condition $k = 2$

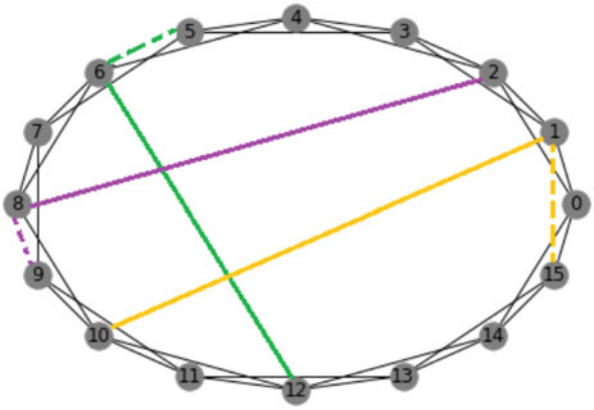


Fig. 3.5 Complex network with three links reconnected. (Reconnections in solid colors, and the original positions in dashed)

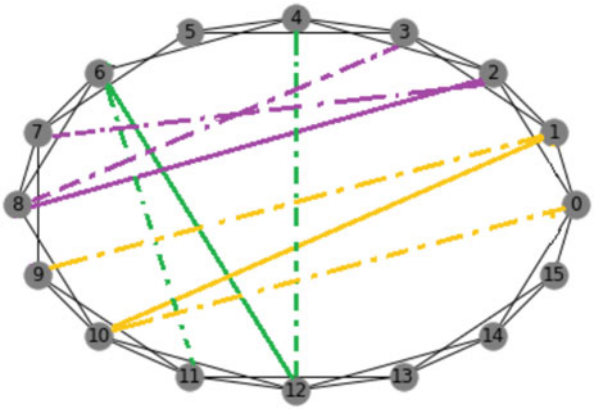


Fig. 3.6 Complex network with three links reconnected, and two more created for any reconnected link

all neighbors of the node that sent the connection are also neighbors of the node that received the connection, the adding should be omitted.

– A connection is created between the node that sent the connection and one of the k neighbors of the node that received the initial connection, chosen randomly. If all the neighbors of the node that receive the connection are also neighbors of the node that sent the connection, the adding should be omitted.

Figure 3.6 presents the complex network in a small-world topology, considering $N = 16$, $K = 2$, and $Nkp = 3$ long-range connections. For more detailed information, refer to [19].

3.3 Chaotic Oscillators

In the scientific literature, there is a wide variety of oscillators reported. However, there is particular interest in those of strange attractors. In this chapter, the Chua oscillator [23], which has been extensively studied, is considered in the nodes of complex networks to be analyzed [24].

3.3.1 Chua Chaotic Oscillator

The Chua oscillator is a nonlinear model capable of generating chaotic behavior that has become a paradigm for studying chaos. The Chua oscillator has been studied extensively because of its appearance, and different generalizations have been proposed. Of the two slopes followed to obtain a generalization of the Chua

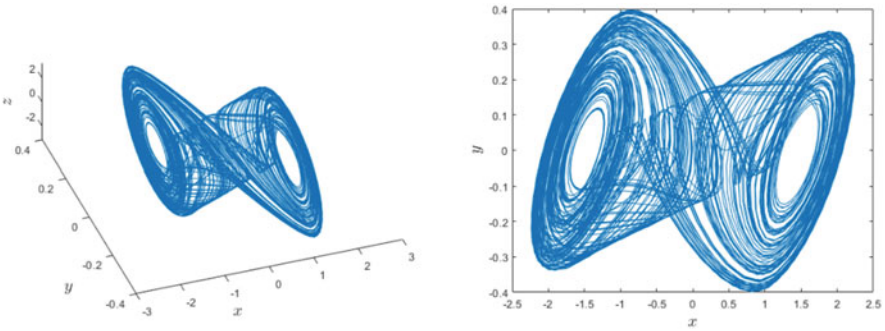


Fig. 3.7 Example of a two-scroll Chua chaotic oscillator. Obtained for the following parameters $\alpha = 9$, $\beta = 14.2857$, $q = 1$, $m = [\frac{-1}{7}, \frac{2}{7}]$

oscillator, one of the most studied is the introduction of additional breakpoints in nonlinearity. The mathematical model, which describes the oscillator generalized chaotic Chua, is given by the following set of differential equations:

$$\begin{aligned} \dot{x} &= \alpha\,[y - h(x)]\,, \\ \dot{y} &= x - y + z, \\ \dot{z} &= -z\beta y, \end{aligned} \tag{3.5}$$

$$h(x) = m_{2q-1}x + \frac{1}{2}\sum_{i=1}^{2q_{11}}(m_{i-1} - m_i)(|x + c_i| - |x - c_i|), \tag{3.6}$$

$$m = \left[m_0,\ , m_1\ , m_2\ , \ldots\ , m_{2q-1}\right], \tag{3.7}$$

$$c = \left[c_0,\ , c_1\ , c_2\ , \ldots\ , c_{2q-1}\right]. \tag{3.8}$$

This is the kind of oscillator we will be using in this chapter. We will assume that every network node has one Chua's system, all of them with the same parameters and different initial conditions. Figure 3.7 shows the phase portraits of the Chua system given in (3.5). With the following parameters $\alpha = 9$, $\beta = 14.2857$, $q = 1$, $m = [\frac{-1}{7}, \frac{2}{7}]$.

3.4 Synchronization

Synchronization refers to the process where two or more systems adjust their movement to a common behavior due to coupling or forcing.

3.4.1 Preliminaries of Small-World Network Synchronization

Consider a dynamic network composed of N identical oscillators, which are linearly coupled, and each one is a dynamic system of dimensions n. The equations of state of the network are given by

$$\dot{x}_i = f(x_i) + c\sum_{j=1}^{N} a_{ij}\Gamma(x_j), \quad i = 1, 2, \ldots, N, \tag{3.9}$$

where $x_i = [x_i^1, x_i^2, \ldots, x_i^n]^T \in \mathbb{R}^n$ is the vector, and $f(\cdot)$ is generally a nonlinear function satisfying a local or global Lipschitz condition. The constant $c > 0$ represents the coupling strength, and $\Gamma : \mathbb{R}^n \to \mathbb{R}^n$ is the inner coupling matrix connection relating the coupled state variables, and $[a_{i,j}] \in \mathbb{R}^{N\times N}$ is the matrix defined as in (3.3) as well called Laplacian matrix.

Mathematically, network synchronization is defined as follows [25].
Network (3.9) is said to achieve complete (asymptotic) synchronization, if

$$\lim_{t\to\infty} ||x_i(t) - x_j(t)|| = 0. \quad \text{for all } 1, 2, \ldots, N, \tag{3.10}$$

where $||\cdot||$ is the Euclidian norm.

Furthermore, the set $\{x_1 = x_2 = \cdots = x_N\}$ is referred to as the synchronization manifold in the state space $\mathbb{R}^{n\times N}$ [1]. Complete synchronization consists of a perfect coupling of the trajectories of the systems. To achieve the complete synchronization of the network given in (3.9) is enough to guarantee that its synchronization manifold be stable invariant synchronization manifold in that for all network orbits in $R^{n\times N}$. Then we can infer that the trajectories are moving to be close enough to synchronization manifold; as a consequence, they will attracted to the manifold and then stay inside forever.

Approximate synchronization results from an asymptotic bound of the difference of a set of variables from one system and the corresponding set from another system.

$$\lim_{t\to\infty} \left|||x(t) - \hat{x}(t)\right|| < \rho, \quad \rho > 0. \tag{3.11}$$

3.4.2 Eigenratio as a Synchronizability Measure

Synchronizability of a dynamical network can be defined as the facility with which the network synchronizes its activity. There is no single interpretation of synchronizability, and several criteria are used to determine it. This concept of synchronizability is related to λ_N/λ_2 and is known as eigenratio, where λ_N and λ_2 are the smallest and largest eigenvalues of the coupling matrix, respectively. In other words, the lower the ratio λ_N/λ_2, the better the synchronizability of the network.

This definition of synchronizability has been extensively used in the literature, being the most frequent expression applied for analyzing the synchronizability of dynamical networks [7, 12].

3.5 Results

Synchronization phenomena have immense applications in science and engineering. In some scenarios, it is desirable to have networks with high levels of synchronization to have better functionality or get better insights into the system. In order to show the effectiveness of the Sanchez–Posadas approach. We measure the synchronizability in small-world complex networks generated by the Watts–Strogatz, Newman–Watts, and Sanchez–Posadas algorithms.

Figures 3.8, 3.9, and 3.10 show the first study case to be analyzed. Here we consider the eigenratio against probability; we maintain the same periodic

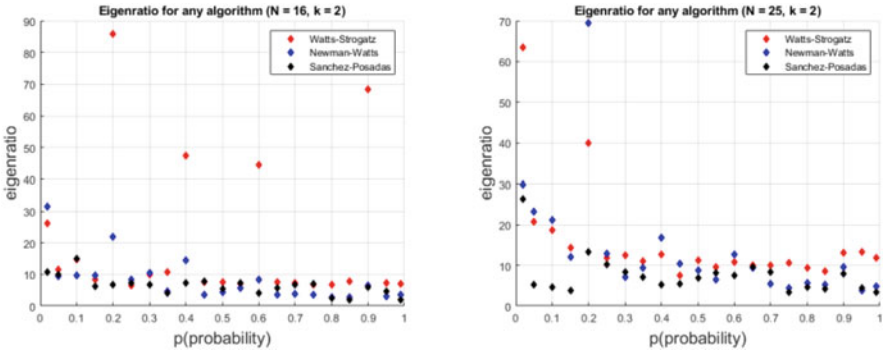


Fig. 3.8 Evolution of eigenratio parameter versus probability for a network of 16 and 25 nodes and connection constant $k = 2$

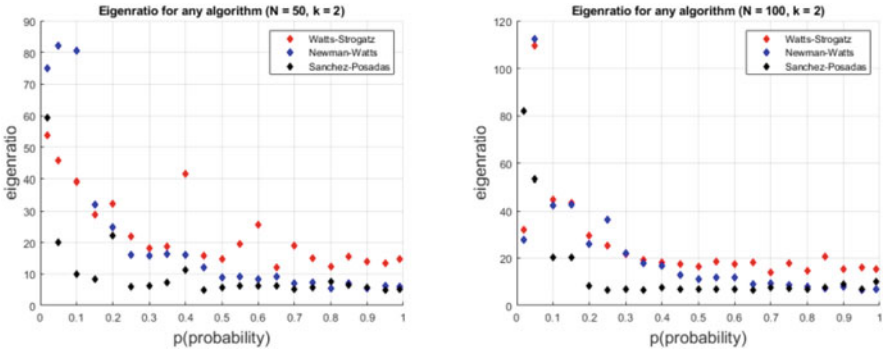


Fig. 3.9 Evolution of eigenratio parameter versus probability for a network of 50 and 100 nodes and connection constant $k = 2$

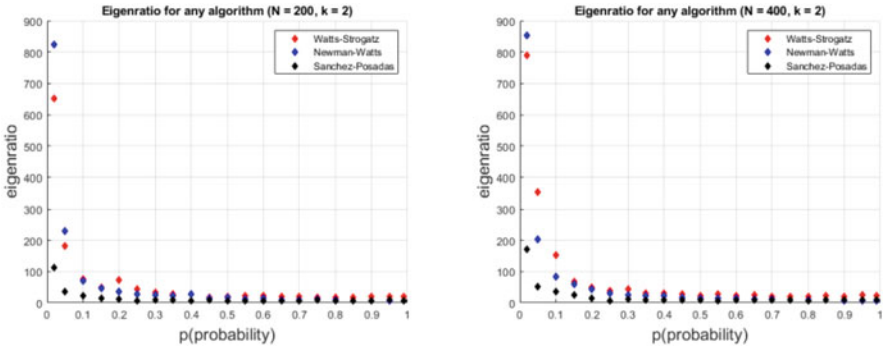


Fig. 3.10 Evolution of eigenratio parameter versus probability for a network of 200 and 400 nodes and connection constant $k = 2$

connectivity and vary the number of nodes in the network as follows $k = 2$ and $N = 16, 25, 50, 100, 200$ and 400, respectively.

It can be seen from Figs. 3.8, 3.9, and 3.10 that as we increase the probability of generating connections in the network for the same number of nodes, the value of the eigenratio decreases, this being an indicator of better synchronizability in said network. In the case of the networks generated with the Sanchez–Posadas algorithm, it can be seen how the value of the eigenratio remains lower even for a relatively small number of nodes, showing a better behavior with respect to the other two studied.

In a second study case, the experiment was performed considering the behavior of the eigenratio. It is compared with the number of nodes in the network by setting the same probability value. The resulting values are shown in Figs. 3.11, 3.12, 3.13, and 3.14. Similar behavior is observed. The Sanchez–Posadas algorithm presents better relation of synchronizability than the other algorithms (Watts–Strogatz and Newman–Watts).

Finally, we proceed to check if the networks generated by these algorithms could achieve synchronization in an emergent way; and with which of the algorithms, the best results would be obtained. To obtain the coupling force value c, the relationship $\lambda_2 = -\frac{T}{c}$ proposed in [20] was used, where T is the instant of time where the states of the chaotic oscillator achieve the stability, using the feedback from state x. As we can see in Fig. 3.15, the system stabilizes at $T = 11s$ approximately.

Figure 3.16 considers a small-world network with 50 nodes and generated with Watts–Strogatz algorithm, it reaches the synchronization of all states considering the Chua's chaotic oscillator in their nodes and a coupling force $c = 17$.

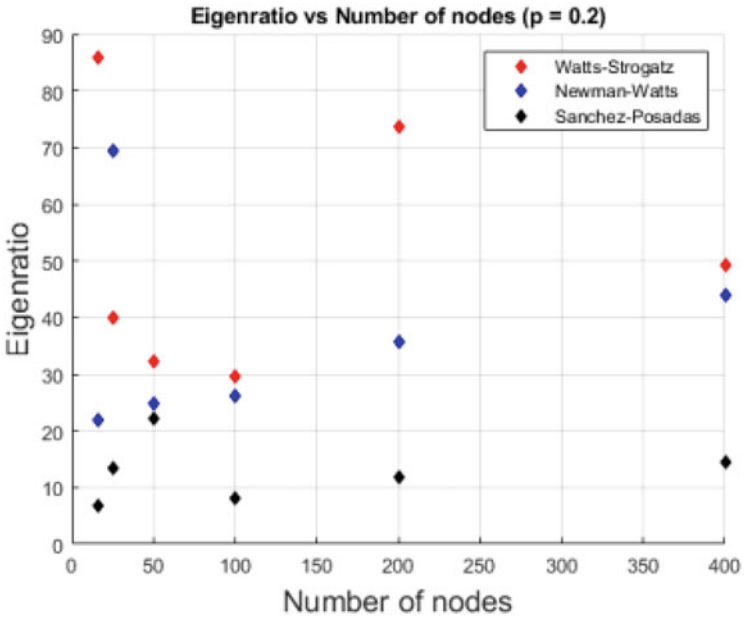


Fig. 3.11 Evolution of eigenratio parameter versus the number of nodes, for a network with probability of reconnection $p = 0.2$

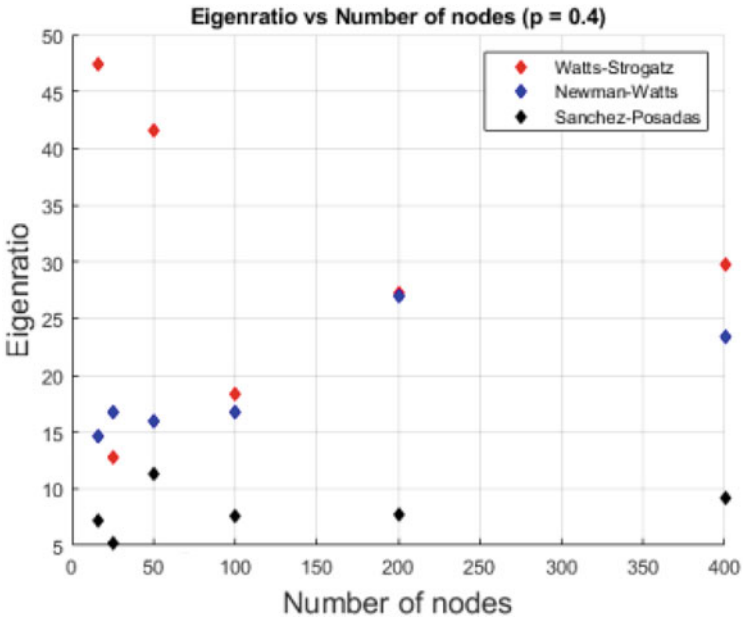


Fig. 3.12 Evolution of eigenratio parameter versus the number of nodes, for a network with probability of reconnection $p = 0.4$

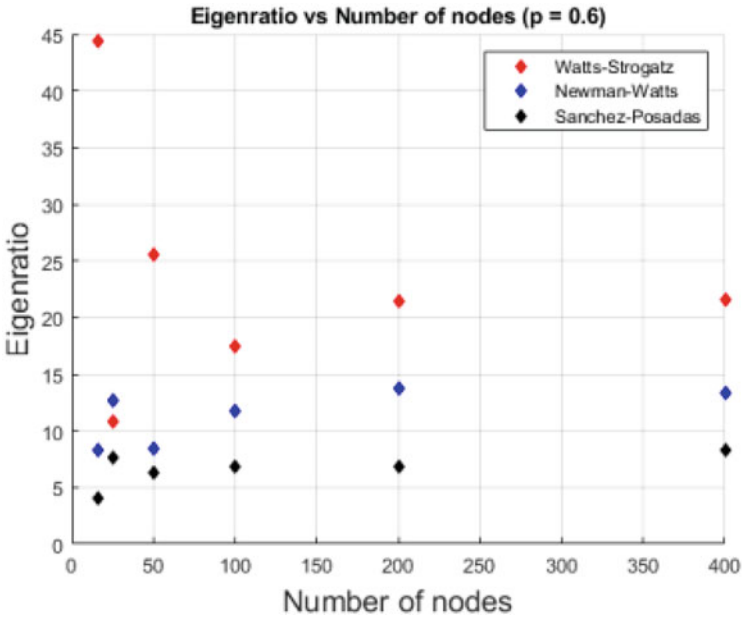


Fig. 3.13 Evolution of eigenratio parameter versus the number of nodes, for a network with probability of reconnection $p = 0.6$

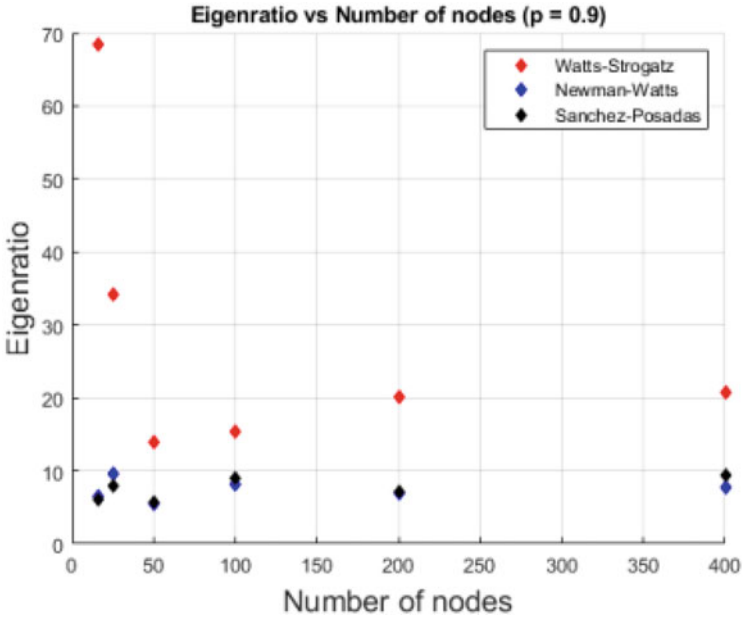


Fig. 3.14 Evolution of eigenratio parameter versus the number of nodes, for a network with probability of reconnection $p = 0.9$

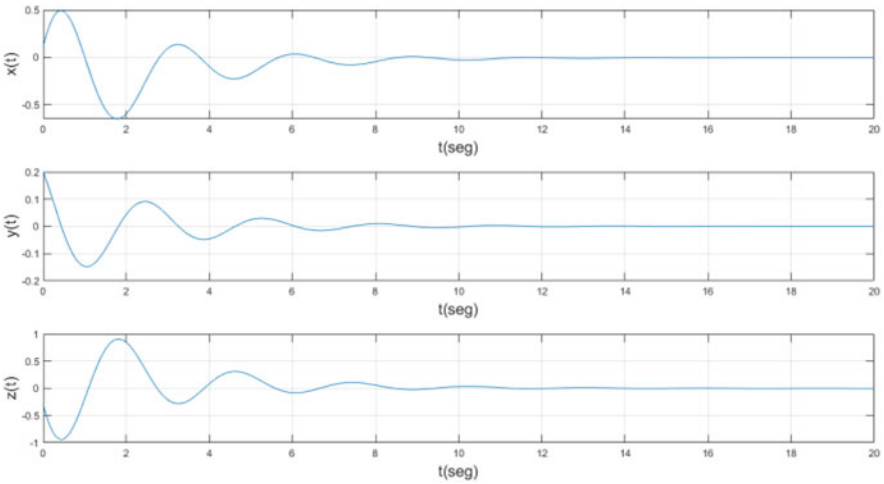


Fig. 3.15 Stabilization of the states Chua's chaotic oscillator

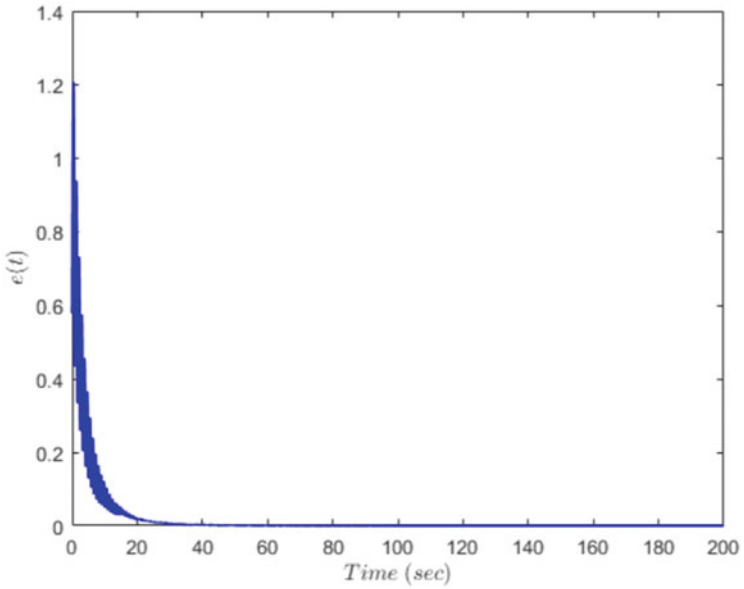


Fig. 3.16 Synchronization with Watts–Strogatz algorithm and synchronization error, for 50 nodes

Figure 3.17 shows the results obtained from a network generated by the Newman–Watts algorithm. It reaches synchronization but presents a small error. This is because synchronization is reached in an emergent way and no control is being applied that forces it to the states to synchronize. It can be said that in this

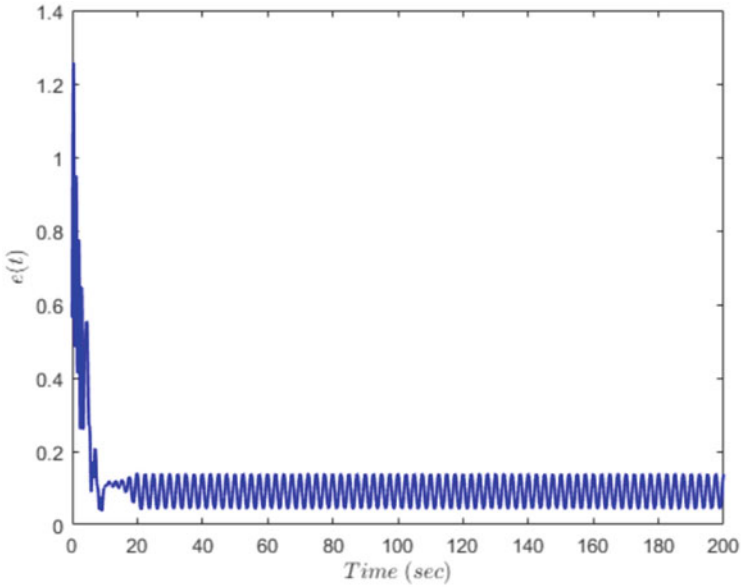


Fig. 3.17 Synchronization with Newman–Watts algorithm and synchronization error, for 50 nodes

case synchronization is achieved but in an approximate type. The coupling force required to achieve this was $c = 6$.

Finally, for the small-world network of Fig. 3.18, which was generated through the Sanchez–Posadas algorithm, it can be observed that it achieves the identical synchronization with the advantage of having achieved it with a lower coupling force than the previous ones. In this case, a coupling force $c = 4$ was required.

Figures 3.19, 3.20, and 3.21 show the synchronized states of each chaotic oscillator, which corresponds to each node of the small-world network generated by Sanchez–Posadas algorithm considering 50 nodes and $c = 4$.

3.6 Conclusions

This chapter presented a comparative study between three algorithms generating complex small-world networks, Watts–Strogatz, Newman–Watts, and Sanchez–Posadas. This comparison was made by generating different small-world networks with each algorithm and observing how the eigenratio parameter varied since this parameter is closely related to the synchronization capacity of complex networks. As a result of this comparison, it was obtained that the small-world complex networks generated by the Sanchez–Posadas algorithm present lower eigenratio values compared to those obtained using the other two algorithms. This means that the Sanchez–Posadas algorithm provides complex small-world networks that

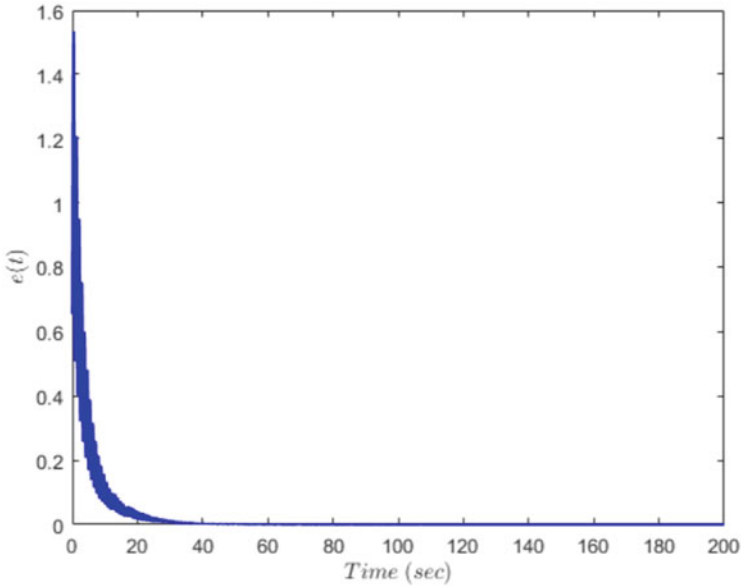


Fig. 3.18 Synchronization with Sanchez–Posadas algorithm and synchronization error, for 50 nodes

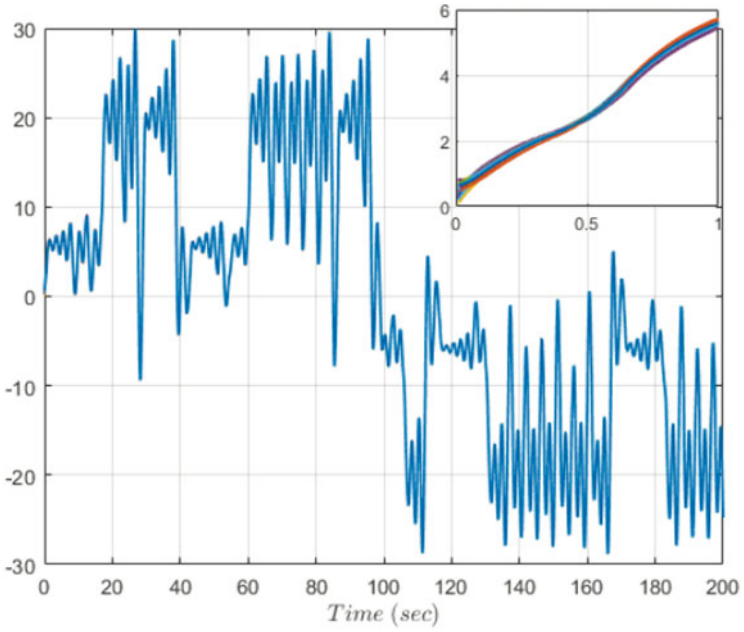


Fig. 3.19 Synchronized x states of each chaotic oscillator

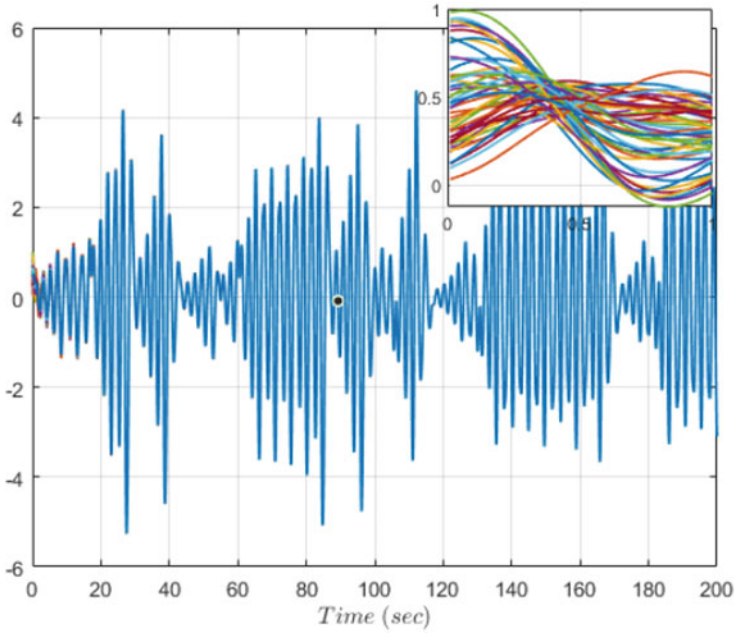


Fig. 3.20 Synchronized y states of each chaotic oscillator

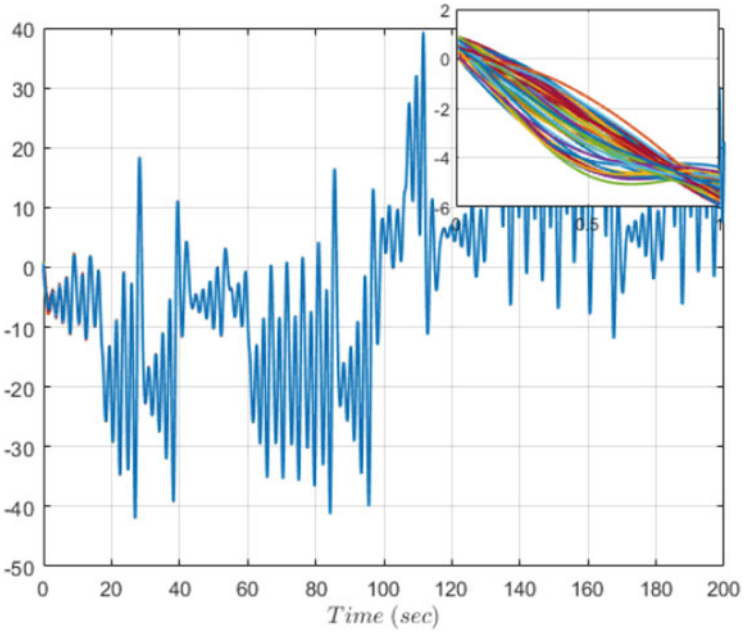


Fig. 3.21 Synchronized z states of each chaotic oscillator

are more prone to synchronization in an emergent manner. This is because a more significant number of connections are introduced for lower probabilities of reconnection with this algorithm. This algorithm is interesting since the networks it generates naturally present good behavior when synchronizing nonlinear dynamic systems.

Acknowledgments All the authors acknowledge to the anonymous reviewers and "Departamento de Electrónica y Automatización" from "Facultad de Ingeniería Mecánica y Eléctrica at UANL. E. Zambrano-Serrano acknowledges CONACYT/Mexico (350385). C. Posadas-Castillo acknowledges CONACYT/Mexico (research grant no. 166654 A1-S-31628).

References

1. G. Chen, X. Wang, X. Li, *Introduction to complex networks: models, structures and dynamics* (Higher Education Press, Beijing, 2012)
2. C.W. Wu, *Synchronization in complex networks of nonlinear dynamical systems* (World Scientific, Singapore, 2007)
3. E. Zambrano-Serrano, J.M. Munoz-Pacheco, A. Anzo-Hernández, O.G. Félix-Beltrán, D.K. Guevara-Flores, Synchronization of a cluster of β-cells based on a small-world network and its electronic experimental verification. Eur. Phys. J. Spec. Top. 1–13 (2021)
4. A. Ruiz-Silva, Synchronization patterns on networks of pancreatic β-cell models. Phys. D Nonlinear Phenom. **416**, 132783 (2021)
5. A. Anzo-Hernández, M. García-Martínez, E. Campos-Cantón, L.J. Ontañón-García, Electronic implementation of a dynamical network with nearly identical hybrid nodes via unstable dissipative systems. Chaos Solitons Fract. **127**, 272–282 (2019)
6. A. Ruiz-Silva, H.E. Gilardi-Velázquez, E. Campos, Emergence of synchronous behavior in a network with chaoticmultistable systems. Chaos Solitons Fract. **151**, 111263 (2021)
7. M. Jalili, Enhancing synchronizability of diffusively coupled dynamical networks: a survey. IEEE Trans. Neural Netw. Learn. Syst. **24**(7), 1009–1022 (2013)
8. A.G.S. Sánchez, C. Posadas-Castillo, E. Garza-González, Determining efficiency of small-world algorithms: a comparative approach. Math. Comput. Simul. **187**, 687–699 (2021)
9. L.M. Pecora, T.L. Carroll, Master stability functions for synchronized coupled systems. Phys. Rev. Lett. **80**(10), 2109 (1998)
10. V.N. Belykh, I.V. Belykh, M. Hasler, Connection graph stability method for synchronized coupled chaotic systems. Phys. D Nonlinear Phenomena **195**(1–2), 159–187 (2004)
11. F. Parastesh, H. Azarnoush, S. Jafari, B. Hatef, M. Perc, R. Repnik, Synchronizability of two neurons with switching in the coupling. Appl. Math. Comput. **350**, 217–223 (2019)
12. H. Gao, J. Zhu, X. Li, X. Chen, Synchronizability of multi-layer-coupled star-composed networks. Symmetry **13**(11), 2224 (2021)
13. Z. Duan, G. Chen, L. Huang, Complex network synchronizability: analysis and control. Phys. Rev. E **76**(5), 056103 (2007)
14. M.S. Tahaei, M. Jalili, M.G. Knyazeva, Synchronizability of EEG-based functional networks in early Alzheimer's disease. IEEE Trans. Neural Syst. Rehabil. Eng. **20**(5), 636–641 (2012)
15. M. Barahona, L.M. Pecora, Synchronization in small-world systems. Phys. Rev. Lett. **89**(5), 054101 (2002)
16. D.J. Watts, S.H. Strogatz, Collective dynamics of 'small-world' networks. Nature **393**(6684), 440–442 (1998)
17. M.E.J. Newman, D.J. Watts, Renormalization group analysis of the small-world network model. Phys. Lett. A **263**(4–6), 341–346 (1999)

18. R. Kasturirangan, Multiple scales in small-world graphs (1999, preprint). arXiv cond-mat/9904055
19. A.G. Soriano-Sánchez, C. Posadas-Castillo, Smart pattern to generate small-world networks. Chaos, Solitons Fract. **114**, 415–422 (2018)
20. A.G. Soriano-Sánchez, C. Posadas-Castillo, M.A. Platas-Garza, A. Arellano-Delgado, Synchronization and FPGA realization of complex networks with fractional-order Liu chaotic oscillators. Appl. Math. Comput. **332**, 250–262 (2018)
21. X.F. Wang, G. Chen, Synchronization in small-world dynamical networks. Int. J. Bifurcation Chaos **12**(1), 187–192 (2002)
22. X.F. Wang, Complex networks: topology, dynamics and synchronization. Int. J. Bifurcation Chaos **12**(5), 885–916 (2002)
23. L.O. Chua, C.W. Wu, A. Huang, G.-Q. Zhong, A universal circuit for studying and generating chaos. I. routes to chaos. IEEE Trans. Circuits Syst I: Fundam. Theory Appl. **40**(10), 732–744 (1993)
24. S. Yu, J. Lu, G. Chen, A novel multiscroll chaotic system and its realization, in *2008 IEEE International Symposium on Circuits and Systems* (IEEE, Piscataway, 2008), pp. 2390–2393
25. X.F. Wang, G. Chen, Synchronization in small-world dynamical networks. Int. J. Bifurcation Chaos **12**(1), 187–192 (2002)

Part II
Cryptosystems

Chapter 4
Generation of Dynamical S-Boxes via Lag Time Chaotic Series for Cryptosystems

B. B. Cassal-Quiroga, A. Ruiz-Silva, and E. Campos-Cantón

Abstract Advances in computing and electronic transactions have led to the need to develop new techniques to ensure information security. In this sense, cryptographic algorithms protect information against any attack. Substitution boxes for block ciphers play an important role as they ensure the security of the algorithms. The aim of this work is to present an algorithm for generating substitution boxes based on series with delay of logistic mapping. By analyzing the dynamics of the logistic mapping, the amount of delays to eliminate the trace of the logistic mapping and the characteristic probability distribution of this system are determined. The performance of the proposed substitution boxes is evaluated using the criteria of a good substitution box, mainly defined by Webster and Tavares. Finally, the proposed substitution boxes are implemented in the image coding to obtain a uniform distribution of pixels.

Keywords S-box · Block cipher · Dynamical S-box · Chaos · Lag time chaotic series

4.1 Introduction

We now live in a technological age that is highly dependent on data and information shared by public channels. This information is indispensable and must be adequately

B. B. Cassal-Quiroga (✉)
Facultad de Ingeniería, Universidad Panamericana, Aguascalientes, Aguascalientes, México
e-mail: bahiacassal@gmail.com

A. Ruiz-Silva
Departamento de Matemáticas, Universidad de Sonora, Hermosillo, Sonora, México
e-mail: adriana.ruizsilva@unison.mx

E. Campos-Cantón
División de Matemáticas Aplicadas, Instituto Potosino de Investigación Científica y Tecnológica A. C., San Luis Potosí, San Luis Potosí, México
e-mail: eric.campos@ipicyt.edu.mx

G. Huerta Cuellar et al. (eds.), *Complex Systems and Their Applications*,
https://doi.org/10.1007/978-3-031-02472-6_4

protected from attackers who can make malicious use it maliciously. To prevent malicious use, the information must be protected in several ways: Confidentiality, Integrity, and Authentication. Cryptography is the science that responds to all these needs in today's communication systems. The two main types of encryption: asymmetric ciphers are based on the use of two keys (public and private) and symmetric ciphers use the same key for encryption and decryption. Symmetric ciphers are again divided into two classes, namely stream ciphers and block ciphers. In stream cipher, one bit of the pseudo-random number generator is combined with one bit of the plaintext, resulting in a bit sequence. In block cipher, blocks of plaintext are encrypted using an S-box and a cyclic shift. The substitution box (S-box) is the core component of the block cipher. The S-boxes give the cryptosystems the confusion property described by Shannon [1], which is used in conventional block ciphers such as the Data Encryption Standard (DES) and the Advanced Encryption Standard (AES). In these cryptosystems, security mainly depends on the properties of the S-box. The criteria that a strong S-box must meet, also known as "good S-boxes," are: Bijection, Nonlinearity, Strict Avalanche Criterion (SAC), and the Output Bit Independence Criterion (BIC) [2]. Other desirable properties include resistance to linear and differential cryptanalysis attacks. The construction of cryptographically secure S-boxes is an interesting area in the field of cryptography.

In recent years, many papers have been published to study cryptosystems based on chaos [3–12] because there is a relationship between the properties of chaotic systems and the properties of cryptosystems. In [13], the relationship between these properties is given, e.g., the confusion is related to the ergodicity, the diffusion property is related to the sensitivity to the initial conditions, and the deterministic dynamics is related to the deterministic pseudo-randomness. Exploiting the properties of chaotic systems, a strong S-box is proposed that satisfies the criteria for good S-boxes.

As for the generation of S-boxes based on chaos, some algorithms have been developed using discrete dynamical systems. For example, in [3–7], the generation of substitution boxes by a single time series of a map or by combining two time series of different maps was presented. However, these algorithms do not guarantee that the series used have a uniform distribution, unlike the approach used in this work. Similarly, there are algorithms based on continuous chaotic dynamical systems [8–10]. There are also algorithms based on mixing time series of continuous and discrete dynamical systems [11, 12] and in [14] the algorithm is built over time-delay series.

The advantage of using discrete chaotic dynamical systems is that the elements of the time series are decorrelated from one iteration to the next. However, when using a continuous chaotic dynamical system, this is not the case because the elements of the time series are highly correlated. Therefore, many iterations are required and the computation of the mutual information between the elements of the time series is necessary to be able to tell when they are decorrelated, which implies higher computational costs.

In chaos-based encryption systems, pseudo-random sequences based on chaotic maps are usually used as a one-time pad for encrypting messages. Since encryption systems based on low dimensional chaotic map have low computational complexity, they can be analyzed with low computational cost using iteration and correlation functions [15]. Time-delay chaotic series have complex behavior and erase the trace of the map that generates them. Using these time series, S-boxes can be developed that provide a better criterion for nonlinearity and ensure good statistical properties of the generators.

In this chapter, the advantage of a cryptographically secure pseudo-random number generator based on the generation of two delayed time series from the logistic map (proposed in [16]) is used to develop a method to obtain dynamically good S-boxes. This type of delayed time series makes possible to generate S-boxes capable of hiding the map used to generate them. Using this approach, pseudo-random series with good statistical properties are generated based on these delayed time series. This novel algorithm for generating S-boxes is based on a cryptographically secure pseudo-random number generator. The rest of the chapter is organized as follows: Sect. 4.2 describes the criteria for a "good" $n \times n$ bit S-box. In Sect. 4.3, a dynamic analysis of the logistic map is presented. In Sect. 4.4, the proposed scheme for generating a dynamic S-box based on a pseudo-random bit generator is presented. In Sect. 4.5, the performance analysis of a obtained S-box and its comparison with other S-boxes described in the literature is presented. An application of the obtained S-boxes to hide an image is presented in Sect. 4.6. Finally, conclusions are drawn in Sect. 4.7.

4.2 Criteria for a Good n×n Bit S-Box

A collection of six criteria reported in the literature for generate cryptographically good S-boxes has been made. These criteria are: bijective, nonlinearity, strict avalanche criterion, output bits independence criterion, equiprobable input/output XOR distribution, and maximum expected linear probability. Before addressing these properties it is necessary to give some preliminaries about Boolean functions.

Let $\mathbb{B} = \{0, 1\}$ be a binary set which is endowed with two binary operations, called addition (denoted by $\oplus$ XOR operation) and multiplication (denoted by $\cdot$ AND operation). Let $(\mathbb{B}, \oplus, \cdot)$ be a field which will be denoted by $\mathbb{F}$, where the binary operations are given by Table 4.1. An $n \times n$ S-box is a vectorial Boolean

Table 4.1 Addition and multiplication binary operations

Input		Output	
A	B	A⊕B	A·B
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

function $S : \mathbb{F}^n \to \mathbb{B}^n$, where $\mathbb{F}^n$ is a vectorial space and is defined as:

$$S(x) = (f_1(x), f_2(x), \cdots, f_n(x)), \tag{4.1}$$

where $x = (x_1, x_2, \cdots, x_n)^\top \in \mathbb{F}^n$ and each of f_i's for $1 \leq i \leq n$ is a Boolean function. A Boolean function is a mapping $f : \mathbb{F}^n \to \mathbb{B}$ by considering all inputs in f, f_i can be seen as a column vector of 2^n elements. The functions f_i's are component functions of S. Some basic definitions can be found in [17].

Definition 1 A Boolean function with algebraic expression, where the degree is at most one is called an **affine Boolean function**. The general form for n-variable affine function is:

$$f_{affine}(x_1, x_2, x_3, \ldots, x_n) = w_n \cdot x_n \oplus w_{n-1} \cdot x_{n-1} \oplus \ldots \oplus w_2 \cdot x_2 \oplus w_1 \cdot x_1 \oplus w_0,$$

where $w_i \in \mathbb{B}$ are coefficients, and $x_i \in \mathbb{B}$ are variables, with $i = 0, 1, \ldots, n$.

Definition 2 A **linear Boolean function** is defined as follows

$$L_w(x) = w_n \cdot x_n \oplus w_{n-1} \cdot x_{n-1} \oplus \ldots \oplus w_1 \cdot x_1,$$

where $x_i, w_i \in \mathbb{B}$, with $i = 1, \ldots, n$.

The set of affine Boolean functions is comprised by the set of linear Boolean functions and their complements, i.e., all functions of the form

$$A_{w,c}(x) = L_w(x) \oplus c, \ c \in \mathbb{B}.$$

A useful representation of a Boolean function f_i, with $i = 1, \ldots, n$, is given by the polarity truth table defined as follows.

Definition 3 A **polarity truth table** is defined as follows

$$\hat{f}(x) = (-1)^{f(x)},$$

where $x \in \mathbb{F}^n$, $\hat{f}$ maps the output values of the Boolean function from the set $\{0, 1\}$ to the set $\{-1, 1\}$, i.e.,

$$\hat{f} : \{0, 1\} \to \{-1, 1\}.$$

A linear Boolean function in polarity form is denoted as $\hat{L}_w(x)$.

Definition 4 The **Walsh Hadamard transform (WHT)** of a Boolean function f is defined as

$$\hat{F}_f(w) = \sum_{x \in \mathbb{B}^n} \hat{f}(x) \hat{L}_w(x).$$

The WHT measures the correlation between the Boolean function f and the linear Boolean function $\hat{L}_w$ with $x \in \mathbb{B}^n$.

4.2.1 Bijective Criterion

Let $S(x)$ be an S-box, which is bijective if and only if their Boolean functions f_i satisfy the following condition:

$$wt(a_1 \cdot f_1 \oplus a_2 \cdot f_2 \oplus \cdots \oplus a_n \cdot f_n) = 2^{n-1}, \tag{4.2}$$

where $a_i \in \mathbb{F}$, $(a_1, a_2, \cdots, a_n) \neq (0, 0, \cdots, 0)$ and $wt(\cdot)$ is the Hamming weight [2, 18], the corresponding S-box is guaranteed to be bijective.

4.2.2 Nonlinearity Criterion

Definition 5 ([19]) The nonlinearity of a Boolean function $f : \mathbb{B}^n \to \mathbb{F}$ is denoted by

$$N_f = \min_{l \in A_{w,c}(x)} d_H(f, l), \tag{4.3}$$

where $A_{w,c}(x)$ is an affine function set, $d_H(f, l)$ is the Hamming distance between f and l.

The minimum distance between two Boolean functions can be described by means of the Walsh spectrum [20]:

$$\min_{l \in A_{w,c}(x)} d_H(f, l) = 2^{n-1}(1 - 2^{-n} \max_{\omega \in \mathbb{F}^n} |\hat{S}_{(f)}(\omega)|), \tag{4.4}$$

where the Walsh spectrum of $f(x)$ is defined as follows:

$$\hat{S}_{(f)}(\omega) = |\hat{F}_f(w)|_{w \in \mathbb{B}^n} = \sum_{x \in \mathbb{F}^n} (-1)^{f(x) \oplus x \bullet \omega}, \tag{4.5}$$

with $\omega \in \mathbb{F}^n$ and $x \bullet \omega$ is the dot product between x and ω as:

$$x \bullet \omega = x_1 \cdot \omega_1 \oplus \cdots \oplus x_n \cdot \omega_n. \tag{4.6}$$

4.2.3 Strict Avalanche Criterion (SAC)

Webster and Tavares gave this criterion, which guarantees that a bit change in the input must change the bits of the output with probability 50% [21]. More formally, a Boolean function f satisfies the condition SAC if and only if

$$\sum_{x \in \mathbb{F}^n} f(x) \oplus f(x \oplus e_i) = 2^{n-1}, \quad \forall i : 1 \leq i \leq n, \tag{4.7}$$

where $e_i \in \mathbb{F}^n$ such that $wt(e_i) = 1$.

4.2.4 Output Bits Independence Criterion (BIC)

Output Bit Independence Criterion is another desirable criterion for an S-box to satisfy, introduced by Webster and Tavares [21]. This means that all avalanche variables should be pairwise independent for a given set of avalanche vectors generated by changing a single plaintext bit. Adam and Tavares introduced another method for measuring BIC, for the Boolean functions f_i and f_j $(i \neq j)$ of two output bits in an S-box, when $f_i \oplus f_j$ is highly nonlinear and as close as possible to satisfying SAC [2]. Moreover, $f_i \oplus f_j$ can be tested with a dynamic distance (DD). Here, the DD of a function f can be defined as:

$$DD(f) = \max_{\substack{d \in \mathbb{F}^n \\ wt(d)=1}} \frac{1}{2} \left| 2^{n-1} - \sum_{x=0}^{2^n-1} f(x) \oplus f(x \oplus d) \right|. \tag{4.8}$$

If the value of DD is a small integer and close to zero, the function f satisfies the SAC.

4.2.5 Criterion of Equiprobable Input/Output XOR Distribution

Differential cryptanalysis, which attacks S-boxes faster than a brute force attack, was introduced by Biham and Shamir [22]. It is desirable for an S-box to have differential uniformity, this can be measured by the maximum expected differential probability (MEDP). The differential probability for a given map S can be calculated by measuring the differential resistance and is defined as follows:

$$DP_f = \max_{\Delta x \neq 0, \Delta y} \left(\frac{\#\{x \in \mathbb{F}^n | S(x) \oplus S(x \oplus \Delta x) = \Delta y\}}{2^m} \right), \tag{4.9}$$

where 2^n is the cardinality of all possible input values (x), Δx and Δy are called input and output differences for the S, respectively. The smaller value of DP_f, the better cryptographic properties, i.e., the resistance to differential cryptanalysis.

4.2.6 Maximum Expected Linear Probability (MELP)

The maximum value of the disequilibrium of an event is the maximum expected linear probability. Given two randomly selected masks a and b, a is used to compute the mask of all possible input values x, and b is used to compute the mask of the corresponding S-box output values. The parity of the input bit mask a is equal to the parity of the output bits of mask b. The MELP of a given S-box can be calculated using the following equation:

$$LP_f = \max_{a,b \in \mathbb{F}^n \setminus \{0\}} \left(2^{-n} \sum_{x \in \mathbb{F}^n} (-1)^{a \cdot x + b \cdot f(x)} \right)^2. \tag{4.10}$$

The closer MELP is to zero, the higher the resistance to linear cryptanalysis attacks.

4.3 Analysis of the Logistic Map

The logistic map is a discrete-time demographic model analogous to the logistic equation first created by Pierre François Verhulst, which is described by the following differential equation

$$\frac{dx}{dt} = rx\left(1 - \frac{x}{K}\right), \tag{4.11}$$

where x is the state variable of the system, r is a parameter related with the rate of maximum population growth and K is the so-called carrying capacity (i.e., the maximum sustainable population). So $x \leq K$, when $x = K$ the population stops growing. Robert May [23] popularized this differential equation to one of the most famous discrete dynamical systems, the logistic map, which is defined as follows:

$$f_\alpha(x_i) = \alpha x_i(1 - x_i), \tag{4.12}$$

where x is the state variable of the logistic map, and α is the only parameter of the system instead of two as its analogous continuous model. The use of a single parameter was possible because the logistic map was normalized, i.e., $f_\alpha : [0, 1] \rightarrow [0, 1]$, for the bifurcation parameter $\alpha \in [0, 4]$ and $x_0 \in [0, 1]$. Nevertheless, in the context of mathematics, the values of the parameter α are not restricted to the

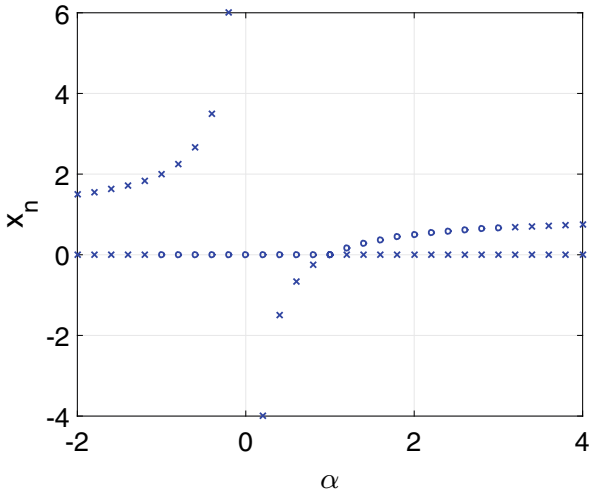


Fig. 4.1 Fixed points stability where an cross and a circle denote repulsive and attracting fixed points, respectively

interval $[0, 4]$, so mathematically, it is possible to consider negative values [24]. As mentioned above, the logistic map is now studied in the interval $[-2, 0)$ for cryptographic purposes. Now, the mapping behavior in the two intervals is analyzed, and with $\alpha \in [-2, 4]$ is assured that the orbits do not scape to infinity for some initial conditions. The dynamical system (4.12) presents one or two fixed points located at $x_1^* = 0$ and at $x_2^* = \frac{\alpha-1}{\alpha}$, for $\alpha \neq 0$. Figure 4.1 depicts the stability of the fixed points where an asterisk and a circle denote repulsive and attracting fixed points, respectively.

These fixed points change their stability according to the parameter α, i.e., when $|f'(x_1^*)| < 1$ and $|f'(x_2^*)| < 1$ then the fixed points x_1^* and x_2^* are stable, respectively, and they are unstable when $|f'(x_1^*)| > 1$ and $|f'(x_2^*)| > 1$. The case of interest is the last, because the system presents complex behavior; this is, both fixed points are repulsive, $|f'(x_1^*)| = |\alpha| > 1$ and $|f'(x_2^*)| = |-\alpha + 2| > 1$.The x_1^* fixed point is repulsive for $\alpha < -1$ and $\alpha > 1$. On the other hand, the x_2^* fixed point is repulsive for $\alpha < 1$ but $\alpha \neq 0$, and $\alpha > 3$. So the interested values are $\alpha \in [-2, -1] \cup [3, 4]$, this is the condition to have both repulsive fixed points. The dynamical system (4.12) bifurcates when $|f'(x_1^*)| = 1$ and $|f'(x_2^*)| = 1$, this happens for x_1^* when $\alpha = -1$ or 1, and for x_2^* the bifurcations values are given by $\alpha = 1$ and 3. It is possible to analyze the behavior of the system by means of a bifurcation diagram, which is shown in Fig. 4.2. This diagram shows orbits as a function of α parameter and the route to chaos are period-doubling bifurcations at $\alpha = 3$ and period-halving bifurcations at $\alpha = -1$. There are intervals for the parameter α near to -2 and 4 where the logistic map $f_\alpha(x)$ behaves chaotically. There are several approaches to demonstrate that a system is chaotic, one of them is prove that the dynamical systems fulfills the definition given by Devaney [25],

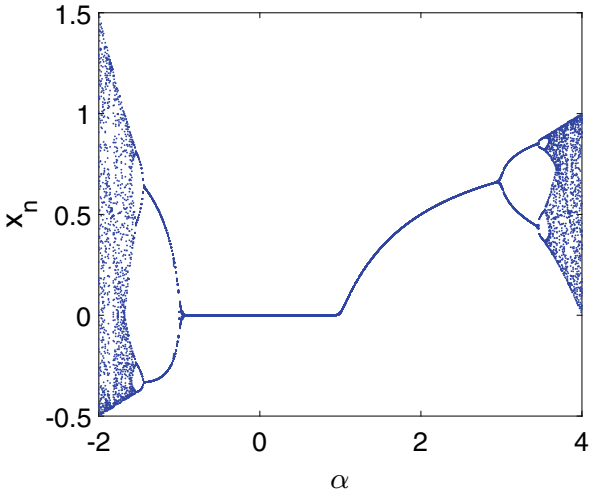


Fig. 4.2 Bifurcation diagram for the logistic map given by Eq. (4.12)

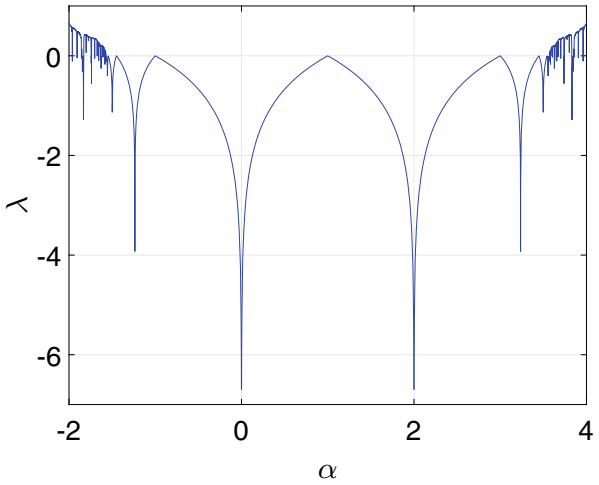


Fig. 4.3 Lyapunov exponent as a function of parameter α

other approach is based on the Lyapunov exponent [26, 27]. The Lyapunov exponent Eq. (4.12) is shown in Fig. 4.3 and it can be seen that the exponents are symmetric with respect to $\alpha = 1$, for values of α near -2 and 4 the chaotic behavior of the logistic map appears. The local stability of the fixed points is consistent with the values of the Lyapunov exponent. For example, when $\alpha \in (-1, 3)$, the orbits of the system converge to a fixed point and when a bifurcation occurs, the orbits converge in periodic orbits until chaos occurs. The goal is to use logistic mapping to produce a uniformly distributed time series without revealing the mapping used.

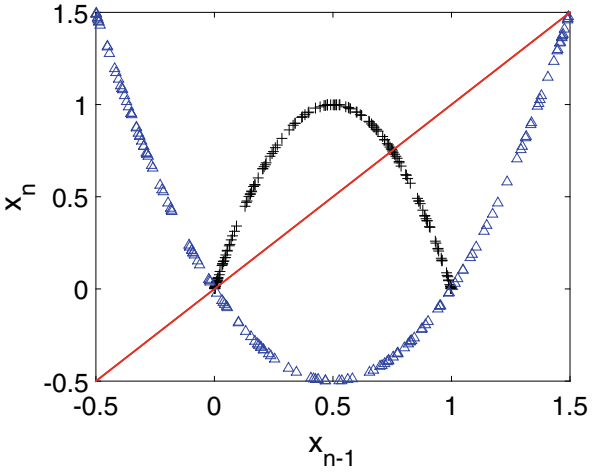


Fig. 4.4 Logistic map for $\alpha = -2$ in blue triangles and for $\alpha = 4$ in black crosses

To achieve this, an approach based on two chaotic time series of the logistic mapping is proposed. Based on Lyapunov exponents analysis, the α values are arbitrarily selected within the chaos region, so it is consider $\alpha = -2$ and 4.

In Fig. 4.4 the shape of the logistic map for the two parameter values $\alpha = -2$ and $\alpha = 4$ is shown in blue triangles and black crosses, respectively. The logistic map for these parameter values is invariant for different intervals as indicated below.

$$\begin{aligned} f_{-2} &: [-0.5, 1.5] \rightarrow [-0.5, 1.5]; \\ f_4 &: [0, 1] \rightarrow [0, 1]. \end{aligned} \tag{4.13}$$

It is worth noting that the time series generated with both parameter values have a U-shaped distribution.

4.4 Proposed Algorithm to Generate Dynamical S-Boxes

The main idea of the proposed algorithm to generate dynamical S-boxes is based on a Cryptographically Secure Pseudo-Random Number Generator (CSPRNG) via a discrete dynamical system $f_\alpha : I \rightarrow I$. García Martínez and Campos Cantón [16] proposed a CSPRNG using two lag time series generated with the logistic map (4.12). An orbit $x_0, x_1, x_2, \ldots$ of the logistic map (4.12) is defined by specifying an initial condition $x_0 \in I$. The parameter $\alpha \in \{-2, 4\}$ determines the interval I, $f_\alpha : I \rightarrow I$. the $M1$ and $M2$ are time series generated by the logistic map using the following considerations: (1) given any two initial conditions x_{01}, x_{02} such that $x_{01} \neq x_{02}$; (2) two different bifurcation parameters α_1 and α_2; and

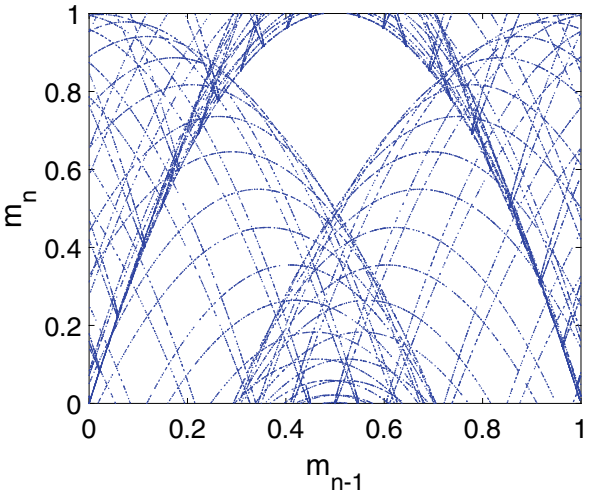


Fig. 4.5 $(m_{(n-1)2}, m_{n2})$ from the time series $M2(x_{(i-k1)2}, x_{i2})$ considering two memory units

(3) l memory units for each time series $x_{(i-k_{l-1})1}, \ldots, x_{(i-k_2)1}, x_{(i-k_1)1}, x_{i1}$ and $x_{(i-k_{l-1})1}, \ldots, x_{(i-k'_2)2}, x_{(i-k'_1)2}, x_{i2}$. Thus, the orbits have a uniform distribution, independent of the U-shaped distribution of the logistic map. To illustrate the algorithm, the bifurcation parameter values $\alpha_1 = -2$ and $\alpha_2 = 4$ are chosen for the time series $M1$ and $M2$, respectively. These parameter values ensure that the system (4.12) exhibits chaotic behavior in both cases. To ensure that the generator has good statistical properties, it is necessary to generate time series with uniform distribution and also it is desirable to eliminate the logistic map shape in these new time series. This is achieved by the number of lags involved. There are many combinations of delays capable of decorrelating the logistic map shape and the time series, but each delay unit requires memory and processing time. If the time series $M2 = m_{02}, m_{12}, m_{22}, \ldots$ are analyzed with two memory units, for $\alpha = 4$ the elements m_{i2} of the time series $M2(x_{(i-k1)2}, x_{i2})$ are obtained in the following way:

$$m_{i2} = M2(x_{(i-k1)2}, x_{i2}) = x_{(i-k1)2} + x_{i2}, \bmod 1, \tag{4.14}$$

where $k1 = 5$. In the plot of $m_{(n-1)2}$ against m_{n2}, it is possible to distinguish that the time series $M2$ are generated with the logistic map. Figure 4.5 shows $(m_{(n-1)2}, m_{n2})$ using two memory units. Because the length of the delay does not matter, the shape of the logistic map always remains, so it is necessary to consider more memory units. By considering three memory units to obtain the elements of the time series $M2(x_{(i-k2)2}, x_{(i-k1)2}, x_{i2})$ given as follows where $k1 = 10$ and $k2 = 5$. For this case of three memory units, which are the minimum amount to obtain cloud of points in $(m_{(n-1)2}, m_{n2})$, see Fig. 4.6. The shape of the logistic map almost

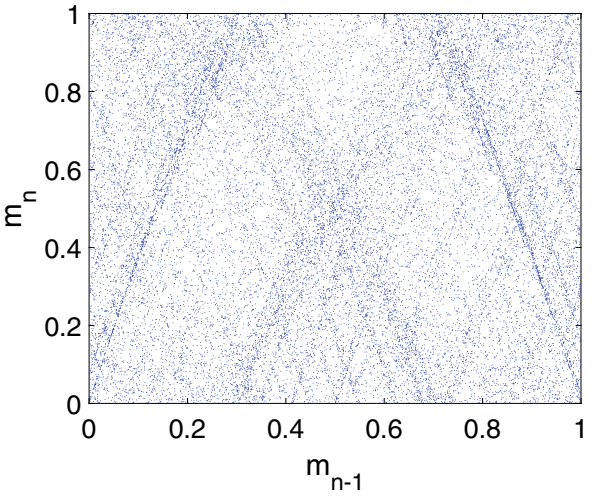


Fig. 4.6 $(m_{(n-1)2}, m_{n2})$ from the time series $M2(x_{(i-k2)2}, x_{(i-k1)2}, x_{i2})$ considering three memory units

disappears, so three memory units are enough:

$$m_{i2} = M2(x_{(i-k2)2}, x_{(i-k1)2}, x_{i2}) = x_{(i-k2)2} + x_{(i-k1)2} + x_{i2}, \text{ mod } 1. \tag{4.15}$$

Also the lags must not be contiguous in order to avoid regular patterns which directly affect the test results. It is considered different delays, i.e., $k_2 = k_2' = 10$, $k_1 = 6$, and $k_1' = 5$ for both time series $M1$ and $M2$. Thus, these series are conformed by the sum of two delay states $x_{(i-10)1}$ and $x_{(i-5)1}$ with the actual state x_{i1} of the orbit $x_{01}, x_{11}, x_{21}, \ldots$, for $M1$. In the same way for $M2$, $x_{(i-10)2}$, $x_{(i-6)2}$ and x_{i2} of the orbit $x_{02}, x_{12}, x_{22}, \ldots$. The values of the time series are limited by the operation mod 1, this guarantees that $M1, M2 \in [0, 1) \subset \Re$. Explicitly $M1(x_{(i-10)1}, x_{(i-5)1}, x_{i1})$ and $M2(x_{(i-10)2}, x_{(i-6)2}, x_{i2})$ are expressed in the following way:

$$m_{i1} = M1(x_{(i-10)1}, x_{(i-5)1}, x_{i1}) = x_{(i-10)1} + x_{(i-5)1} + x_{i1}, \text{ mod } 1, \tag{4.16}$$

$$m_{i2} = M2(x_{(i-10)2}, x_{(i-6)2}, x_{i2}) = x_{(i-10)2} + x_{(i-6)2} + x_{i2}, \text{ mod } 1. \tag{4.17}$$

Finally, these time series $M1 = m_{01}, m_{11}, m_{21}, \ldots$ and $M2 = m_{02}, m_{12}, m_{22}, \ldots$ given by (4.16) and (4.17), respectively, are mixed and the operation mod 1 is applied again, this process generates a new time series Z_i given as follows:

$$Z_i = m_{i1} + m_{i2}, \text{ mod } 1. \tag{4.18}$$

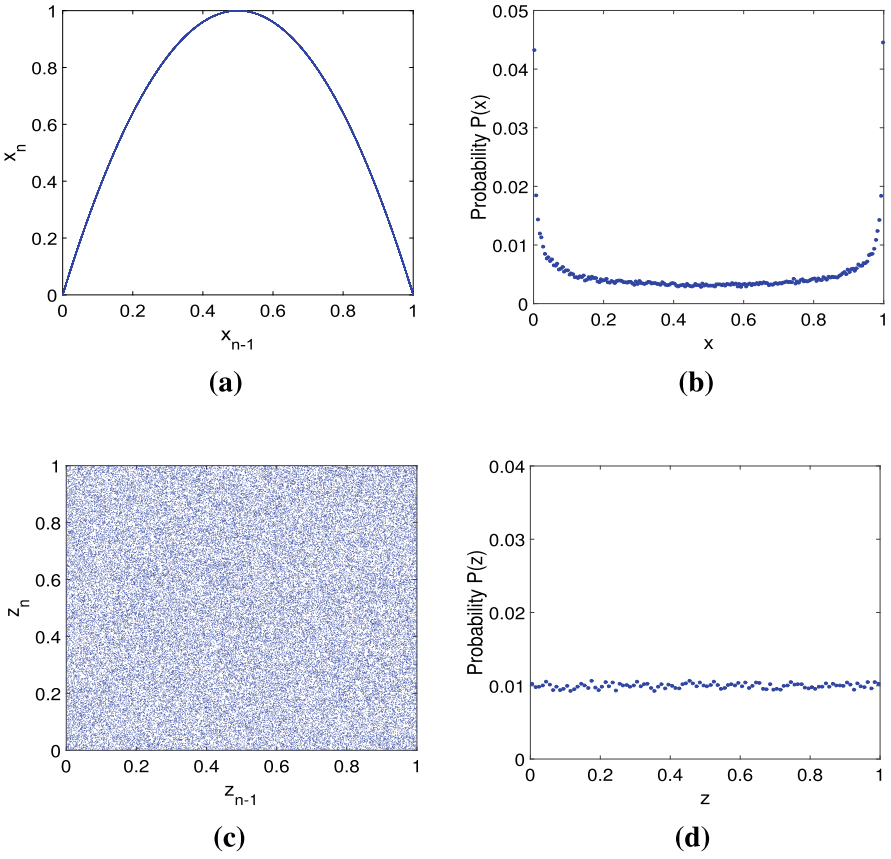


Fig. 4.7 (**a**) Logistic map given by x_n against x_{n-1}; (**b**) "U-shaped" probability distribution of the logistic map; (**c**) Delayed map given by z_n against z_{n-1}; (**d**) Uniform probability distribution of the delayed map

From now on Eq. (4.18) is referred as the delayed map. Note that $Z_i \in [0, 1) \subset \Re$. The aim to use this approach is that with the combination of two time series with delays represented by Z_i, it is possible to dismiss the structure of the chaotic map used. For instance, the time series x_n can reveal the map whether x_n against x_{n-1} is plotted as is shown in Fig. 4.7a, the logistic map appears. In contrast, the time series z_n cannot reveal the map whether z_n against z_{n-1} is plotted as is shown in Fig. 4.7c, the delays used are not reveled neither does the logistic map appear. As well as, this allows us to change the characteristic "U-shaped" probability distribution [28] by a uniform probability distribution in the obtained time series x_n and z_n, see Fig. 4.7b, d, respectively. This is an important characteristic that, in comparison with the chaos-based schemes approach, makes easier the construction of S-box since all values has the same probability of occurrence in contrast with a single chaotic based schemes.

To obtain a binary time series s useful for cryptosystems, it is constructed the symbolic dynamics of Z_i. So the elements of s are binary numbers, i.e., $s_i(Z_i) \in \{0, 1\}$. One necessary requirement for the symbolic dynamic is to obtain zeros or ones with the same probability, thus the process for getting the binary series is as follows:

$$s_i = \begin{cases} 0, & \text{for } 0 < Z_i \leq 0.5, \\ 1, & \text{for } 0.5 < Z_i < 1. \end{cases} \tag{4.19}$$

A CSPRNG based on a discrete dynamical system is given from Eqs. (4.12)–(4.19).

4.4.1 The Algorithm for S-Box Design via CSPRNG

In this subsection, we propose a new algorithm for creating $n \times n$ S-boxes based on CSPRNG. The algorithm is simple and is described step by step as follows:

Step 1 Choose the initial conditions x_{01} and x_{02} for CSPRNG to generate the stream of bits $s_0, s_1, s_2, \ldots$.

Step 2 Generate the block sequence of n-bits each, $C_0 = (s_0, s_1, \ldots, s_{n-1})$, $C_1 = (s_n, s_{n+1}, \ldots, s_{2n-1})$, $C_2 = (s_{2n}, s_{2n+1}, \ldots, s_{3n-1})$, $\ldots$.

Step 3 Convert the blocks $C_0, C_1, C_2, \ldots$ of n-bits to integer $D_0, D_1, D_2, \ldots$.

Step 4 Discard the repeated elements D's to select 2^n distinct values. The rule to discard an element is as follows: if $D_i = D_j$ with $i < j$ then discard D_j.

Step 5 Create the S-Box with the 2^n distinct elements of D's.

Upon completion of the procedure, the proposed algorithm returns a $n \times n$ S-box with unique 2^n values. Note that D_0 is the first element of the S-box, but the second element cannot be D_1 if $D_0 = D_1$. However, enough 2^n elements have been generated to form the S-box. Each block C consists of n bits, $s_j, s_{j+1}, \ldots, s_{j+n-1}$, associated with functions f_i, with $i = 1, \ldots, n$. For example, if $n = 8$, $x_{01} = 0.8147$, $x_{02} = 0.9058$, $\alpha_1 = 4$ and $\alpha_2 = -2$, then the 8×8 S-box in Table 4.2 is obtained.

This proposed substitution box is the only component in a cryptosystem (block ciphers) that results in a nonlinear mapping between inputs and outputs, leading to confusion in the data. In the next section, the performance of the proposed algorithm for generating S-boxes is investigated to confirm their immunity to differential and linear cryptanalysis.

Table 4.2 The S-box generated by proposed algorithm

64	46	150	174	220	26	233	224	148	170	143	247	225	212	90	124
44	204	59	61	43	121	129	2	109	164	103	249	16	237	27	35
216	184	81	213	161	169	89	199	140	38	239	48	163	193	21	147
222	217	70	196	195	192	234	41	47	15	14	42	98	190	186	36
242	51	60	87	24	104	189	55	118	111	231	120	8	226	7	141
85	9	73	101	3	197	12	66	82	110	65	25	165	176	80	181
125	31	218	74	68	52	149	95	182	19	112	5	136	79	214	34
158	50	188	137	28	191	155	84	105	126	92	179	162	152	200	0
171	142	240	203	88	160	32	202	99	18	100	97	145	53	194	93
245	119	185	20	235	123	134	139	128	116	173	76	17	132	209	135
83	168	57	56	223	30	91	4	22	122	102	221	208	131	71	86
39	114	252	10	172	201	177	77	94	246	54	175	183	108	156	45
219	210	40	130	113	153	13	166	58	23	253	215	238	33	198	248
229	227	96	206	107	144	67	254	115	167	244	106	180	157	255	241
207	243	228	187	49	78	251	37	62	1	205	117	29	178	75	236
11	250	146	6	151	69	138	133	72	232	211	127	159	63	154	230

4.5 Performance Test of S-Box

In this section, six important and well-known cryptographic criteria of the 8×8 S-boxes are computed. Lastly, the obtained results are contrasted with some results presented in different published papers using other approaches.

4.5.1 Bijectivity Criterion

The computed value of proposed S-box is the desired value of $2^{n-1} = 128$, with $n = 8$, according to the formula (4.2). So the bijectivity criterion is satisfied and the S-box proposed is a one-to-one, surjective, and balance, which is a primary cryptographic criterion.

4.5.2 Nonlinearity Criterion

If the function has a high minimum Hamming distance, it is said to have high nonlinearity by reducing the Walsh spectrum in (4.4).

Nonlinearity is the most important requirement for the design of an S-box, since it ensures that an S-box is not a linear function between input vectors and output vectors. The nonlinearity symbolizes the degree of dissimilarity between the Boolean function f and the linear n-bit function l. The nonlinearity test used affine

Table 4.3 SAC values of proposed S-box

0.5781	0.4844	0.5000	0.4219	0.4844	0.5156	0.4063	0.5469
0.5156	0.5000	0.4688	0.5156	0.5469	0.3906	0.5469	0.4375
0.5469	0.5000	0.5000	0.5469	0.4063	0.5156	0.4531	0.5313
0.4531	0.5156	0.5000	0.4531	0.5313	0.5313	0.4844	0.4688
0.5156	0.5469	0.4844	0.5313	0.5313	0.5625	0.5625	0.5469
0.4063	0.4844	0.5000	0.4063	0.5625	0.5625	0.4844	0.5313
0.4219	0.4063	0.5313	0.5313	0.4219	0.5625	0.4844	0.4844
0.5469	0.5156	0.5469	0.5625	0.4531	0.5625	0.5781	0.4531

functions, which are cryptographically weak. If the Hamming distance between the Boolean function of the s-box and the affine function is smaller, then the s-box is cryptographically weak. An S-box contains n Boolean functions and the nonlinearity of each Boolean function must be computed. The nonlinearities of the proposed S-box are 104, 104, 102, 104, 96, 102, 100, and 102, respectively. The optimal value of nonlinearity is 120. The high nonlinearity ensures the strongest ability to resist strong modern attacks such as linear cryptanalysis.

4.5.3 Strict Avalanche Criterion (SAC)

The avalanche effect is used to indicate the randomness of an S-box when an input undergoes a change. Accordingly, an S-box whose SAC value is approximately equal to 0.5 is considered strong. The dependency matrix of the SAC values of our S-box can be found in Table 4.3.

For the S-box proposed, it is obtained a maximum SAC equal to 0.5781, the minimum is 0.3906, and its average value 0.5012 is close to the desired value 0.5. Based on these results, it can be concluded that the S-box generated by the proposed method fulfills the property of SAC.

4.5.4 Output Bits Independence Criterion (BIC)

The BIC criterion guarantees that there are no statistical patterns or dependencies between the output vectors. The BIC of the S-box generated by the proposed method is tested as described in Sect. 4.5.4. The obtained results are presented in Tables 4.4, 4.5, and 4.6.

From Table 4.4 the average value comes out to be 103.8571, for Table 4.5 the average value is 0.5066, and the maximum value of Table 4.6 is 8. These values indicate that the projected S-box satisfies the BIC requirement well.

Table 4.4 BIC-nonlinearity of Boolean function $f_i \oplus f_j$ $(i \neq j)$

0	104	104	106	104	106	106	102
104	0	106	98	102	104	102	104
104	106	0	104	102	96	104	104
106	98	104	0	106	100	106	104
104	102	102	106	0	102	100	102
106	104	96	100	102	0	104	108
106	102	104	106	100	104	0	106
102	104	104	104	102	108	106	0

Table 4.5 BIC-SAC of Boolean function $f_i \oplus f_j$ $(i \neq j)$

0	0.5020	0.5176	0.5137	0.5293	0.5098	0.4727	0.5059
0.5020	0	0.4980	0.4844	0.5039	0.5313	0.5156	0.5000
0.5176	0.4980	0	0.5039	0.4941	0.5313	0.5000	0.5020
0.5137	0.4844	0.5039	0	0.5117	0.4980	0.5020	0.5020
0.5293	0.5039	0.4941	0.5117	0	0.5234	0.5000	0.5137
0.5098	0.5313	0.5313	0.4980	0.5234	0	0.5039	0.5000
0.4727	0.5156	0.5000	0.5020	0.5000	0.5039	0	0.5156
0.5059	0.5000	0.5020	0.5020	0.5137	0.5000	0.5156	0

Table 4.6 The DD of the generated S-box (BIC–SAC criterion)

0	2	6	2	4	6	4	6
2	0	2	2	2	2	4	2
6	2	0	6	8	2	6	4
2	2	6	0	4	2	8	4
4	2	8	4	0	2	0	2
6	2	2	2	2	0	2	8
4	4	6	8	0	2	0	0
6	2	4	4	2	8	0	0

4.5.5 Criterion of Equiprobable Input/Output XOR Distribution

The equiprobable Input/Output XOR Distribution is a criterion which analyzes the effect, in particular, differences in input pairs of the resultant output pairs to discover the key bits. The idea is to find the high probability difference pairs for an S-Box under attack. The equiprobable input/output XOR distribution of generated S-box calculated by (4.9) is presented in Table 4.7. Maximal value of S-box generated by the proposed method is 5, which indicates that the S-box satisfies bound for the equiprobable Input/Output XOR Distribution criterion.

Table 4.7 Equiprobable input/output XOR Distribution approach table for the generated S-box

4	3	3	4	3	3	3	3	4	3	3	3	3	3	4	4
3	3	4	3	3	4	3	3	4	3	3	4	4	4	4	3
4	3	4	3	4	4	3	3	3	3	3	3	3	4	3	3
4	3	3	3	4	4	4	4	3	4	5	4	3	2	3	3
5	4	4	3	3	3	4	4	4	3	5	3	3	3	3	3
3	3	3	4	4	3	5	4	3	3	3	5	5	3	3	3
3	3	3	3	3	4	4	3	3	3	4	3	3	2	3	3
3	2	3	3	3	4	3	3	3	3	3	4	3	3	3	3
3	3	3	5	5	3	3	4	3	4	3	2	5	3	3	3
3	3	3	4	3	4	3	3	3	4	3	3	4	3	4	3
4	3	4	3	2	3	3	4	3	3	3	3	3	4	3	3
3	4	3	3	3	3	3	3	3	4	3	3	3	3	4	4
3	3	3	3	3	4	3	3	2	4	3	3	4	4	3	3
4	3	4	3	4	4	3	4	4	3	4	4	3	3	3	3
3	4	3	3	3	3	3	3	3	4	4	3	3	3	3	3
3	3	3	5	4	5	4	3	3	5	3	3	4	3	5	-

4.5.6 MELP Criterion of the Generated S-Box

The MELP value are calculated using linear approximations to model nonlinear steps. The final goal is to recover the key bits or part of the key bits. MELP examines the statistical correlation between the input and the output. This criterion of the proposed S-boxes are calculated according to the Eq. (4.9) and the average value is 0.0716.

4.5.7 Comparative Results

In this section, we present a performance comparison of the S-boxes created using the algorithm described in Sect. 4.4.1. In Table 4.8, we list the criterion values for the proposed S-boxes and a set of widely known boxes (standard and chaos-based S-boxes).

From this Table 4.8, it can be seen that the generated S-boxes satisfy the most important condition, bijectivity, and have a good similarity with the other expected test values [3, 7, 10, 29–36]. It mainly shows better performance on the tests related to attacks (MELP and equiprobable Input/Output XOR Distribution). Moreover, it is a method based on a system with simple operations that generates sequences with complex behavior.

Table 4.8 Comparison of recent chaos-base designed S-boxes and S-boxes used in typical block ciphers

	Bijective	Nonlinearity			SAC			BIC			I/O XOR	MELP
		Min	Max	Avg	Min	Max	Avg	SAC	Nonlinearity	DD		
Skipjack S-box [37]	123	100	108	105.1250	0.3906	0.5938	0.5027	0.5003	104.03	109	0.0469	0.0594
APA S-box [29]	128	112	112	112	0.4375	0.5625	0.5007	0.4997	112	112	0.0156	0.0156
Gray S-box [30]	128	112	112	112	0.4375	0.5625	0.4998	0.5026	112	112	0.0156	0.0156
AES S-box [31]	128	112	112	112	0.4531	0.5625	0.5049	0.5046	112	112	0.0156	0.0156
Ref. [3]	128	98	107	103.25	0.3828	0.5938	0.5059	0.5033	104.21	108	0.0469	0.0665
Ref. [32]	129	103	109	104.875	0.3984	0.5703	0.4966	0.5044	102.96	109	0.0391	0.0706
Ref. [33]	128	96	106	103	0.3906	0.6250	0.5039	0.5010	100.35	106	0.5000	0.0881
Ref. [34]	128	112	112	112	0.4219	0.5469	0.5115	0.4982	108.71	112	0.0313	0.0479
Ref. [7]	128	102	108	105.25	0.4375	0.5781	0.5056	0.5019	103.78	108	0.0391	0.0977
Ref. [10]	128	104	110	106.25	0.4219	0.5938	0.5039	0.5059	103.35	108	0.0391	0.0791
Ref. [35]	128	102	108	106	0.4219	0.5938	0.5002	0.5016	104.42	108	0.0391	0.0881
Ref. [36]-1	128	106	108	106.75	0.3906	0.6094	0.4941	0.5013	104.28	108	0.0391	0.0625
Ref. [36]-2	128	106	108	106.75	0.4063	0.5938	0.4971	0.5008	102.92	106	0.0391	0.0791
The proposed S-box	128	96	104	101.75	0.3906	0.5781	0.5012	0.5066	103.42	108	0.0391	0.0706

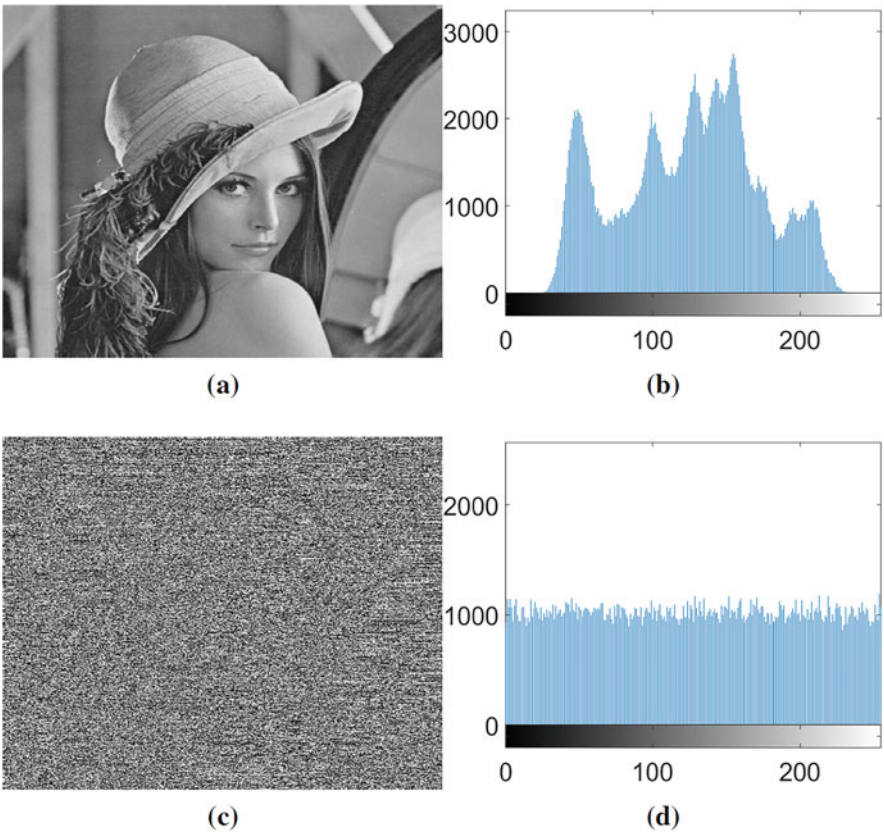


Fig. 4.8 Plain image of Lena, encrypted image and their histograms

4.6 Dynamical Generation of S-Boxes and Its Application

The Alberti cipher was one of the first polyalphabetic ciphers, where the principle is based on substitution, using multiple substitution alphabets so that the output has a uniform distribution. Based on this idea of polyalphabetic ciphers, an application of dynamic generation of S-boxes is presented, i.e., a certain intensity of a pixel can be substituted by other intensities in the same round. Usually, an S-box is used to replace all pixels of an image of size $p \times q$ in the same way. The idea of polyalphabetic ciphers is to use a dynamic S-box to achieve this goal by finding a uniform distribution. Thus, the proposed dynamic S-boxes belong to a class of S-boxes given by p elements (S-boxes) generated by the algorithm presented in the Sect. 4.4.1. The approach to obtain a uniform distribution is to apply a dynamic S-box that changes with each row of pixels. That is, the dynamic S-box is modified by a different S-box for each row. Figure 4.8 shows the original Lena image and the codified Lena image with their gray scale pixel distribution.

In cryptography a uniform distribution is always desired, since this property was achieved by simple substitution with the S-boxes, a good result can be expected for a full cryptographic algorithm based on these S-boxes. It is important to point out that this is not an encryption algorithm, but a simple and useful approach intended to show possible applications for the dynamic S-boxes presented in this work.

4.7 Concluding Remarks

In this work, a simple algorithm has been proposed to generate $n \times n$ S-boxes using a pseudo-random bit generator (PRBG) based on two delayed time series of the logistic map. The mixture of these two time series favors a uniform distribution in addition to the obfuscation of the used chaotic map. In order to evaluate the performance of the proposed S-box, several statistical tests were performed. The results of the numerical analysis of this cryptographically strong S-box generated by the algorithm have also shown that all the criteria for a good S-box have been met and there is a high immunity to differential cryptanalysis and linear cryptanalysis. The performance test result was compared with other S-boxes reported in the literature. Finally, an application based on a simple and useful approach to achieve a uniform distribution was presented.

References

1. C.E. Shannon, Communication theory of secrecy systems. Bell Syst. Tech. J. **28**(4), 656–715 (1949)
2. C. Adams, S. Tavares, The structured design of cryptographically good s-boxes. J. Cryptol. **3**(1), 27–41 (1990)
3. G. Jakimoski, L. Kocarev, Chaos and cryptography: block encryption ciphers based on chaotic maps. IEEE Trans. Circuits Syst. I Fundam. Theory Appl. **48**(2), 163–169 (2001)
4. G. Chen, A novel heuristic method for obtaining s-boxes. Chaos Solitons Fract. **36**(4), 1028–1036 (2008)
5. Y. Wang, K.-W. Wong, X. Liao, T. Xiang, A block cipher with dynamic s-boxes based on tent map. Commun. Nonlinear Sci. Numer. Simul. **14**(7), 3089–3099
6. D. Lambić, A novel method of s-box design based on chaotic map and composition method. Chaos Solitons Fract. **58**, 16–21 (2014)
7. A. Belazi, M. Khan, A.A.A. El-Latif, S. Belghith, Efficient cryptosystem approaches: S-boxes and permutation–substitution-based encryption. Nonlinear Dynam. **87**(1), 337–361 (2017)
8. F. Özkaynak, A.B. Özer, A method for designing strong s-boxes based on chaotic Lorenz system. Phys. Lett. A **374**(36), 3733–3738 (2010)
9. G. Liu, W. Yang, W. Liu, Y. Dai, Designing s-boxes based on 3-d four-wing autonomous chaotic system. Nonlinear Dynam. **82**(4), 1867–1877 (2015)
10. Ü. Çavuşoğlu, A. Zengin, I. Pehlivan, S. Kaçar, A novel approach for strong s-box generation algorithm design based on chaotic scaled Zhongtang system. Nonlinear Dynam. **87**(2), 1081–1094 (2017)

11. R. Guesmi, M.A.B. Farah, A. Kachouri, M. Samet, A novel design of chaos based s-boxes using genetic algorithm techniques, in *2014 IEEE/ACS 11th International Conference on Computer Systems and Applications (AICCSA)* (2014), pp. 678–684.
12. Y. Tian, Z. Lu, S-box: six-dimensional compound hyperchaotic map and artificial bee colony algorithm. J. Syst. Eng. Electron. **27**(1), 232–241 (2016)
13. G. Alvarez, S. Li, Some basic cryptographic requirements for chaos-based cryptosystems. Int. J. Bifurcation Chaos **16**(8), 2129–2151 (2006)
14. F. Özkaynak, S. Yavuz, Designing chaotic s-boxes based on time-delay chaotic system. Nonlinear Dynam. **74**(3), 551–557 (2013)
15. Y. Zhou, L. Bao, C.P. Chen, Image encryption using a new parametric switching chaotic system. Signal Process. **93**(11), 3039–3052 (2013)
16. M. García-Martínez, E. Campos-Cantón, Pseudo-random bit generator based on lag time series. Int. J. Mod. Phys. C **25**(4), 1350105 (2014)
17. D. Souravliasa, K.E. Parsopoulos, G.C. Meletiou, Designing bijective s-boxes using algorithm portfolios with limited time budgets. Appl. Soft Comput. **59**(1), 475–486 (2017)
18. C. Adams, S. Tavares, Good s-boxes are easy to find, in *Advances in Cryptology — CRYPTO'89 Proceedings*, ed. by G. Brassard (Springer New York, 1990), pp. 612–615
19. Y. Tian, Z. Lu, Chaotic s-box: intertwining logistic map and bacterial foraging optimization. Math. Problems Eng. **2017**, 6969312 (2017)
20. W. Millan, How to improve the nonlinearity of bijective s-boxes, in *Information Security and Privacy*, ed. by C. Boyd, E. Dawson (Springer, Berlin, 1998), pp. 181–192
21. A.F. Webster, S.E. Tavares, On the design of s-boxes, in *Advances in Cryptology — CRYPTO'85 Proceedings*, ed. by H.C. Williams (Springer, Berlin, 1986), pp. 523–534
22. E. Biham, A. Shamir, Differential cryptanalysis of DES-like cryptosystems, in *Advances in Cryptology-CRYPT0'90*, ed. by A.J. Menezes, S.A. Vanstone (Springer, Berlin, 1991), pp. 2–21
23. R.M. May, Simple mathematical models with very complicated dynamics. Nature **261**(5560), 459 (1976)
24. D.S. Dendrinos, M. Sonis, Socio-spatial stocks and antistocks; the logistic map in real space. Ann. Regional Sci. **27**(4), 297–313 (1993)
25. R.L. Devaney, *An introduction to chaotic dynamical systems* (Westview Press, Boulder, 2018)
26. C. Li, G. Chen, Estimating the Lyapunov exponents of discrete systems. Chaos **14**(2), 343–346 (2004)
27. C. Yang, C.Q. Wu, P. Zhang, Estimation of Lyapunov exponents from a time series for n-dimensional state space using nonlinear mapping. Nonlinear Dynam. **69**(4), 1493–1507 (2012)
28. J. Urías, E. Campos, N.F. Rulkov, *Random Finite Approximations of Chaotic Maps* (Springer, New York, 2006), pp. 231–242
29. L. Cui, Y. Cao, A new s-box structure named affine-power-affine. Int. J. Innov. Comput. Inf. Control **3**(3), 751–759 (2007)
30. M.T. Tran, D.K. Bui, A.D. Duong, Gray s-box for advanced encryption standard, in *2008 International Conference on Computational Intelligence and Security*, vol. 1 (IEEE, Piscataway, 2008), pp. 253–258
31. J. Daemen, V. Rijmen, AES proposal: Rijndael (1999). http://csrc.nist.gov/archive/aes/rijndael/Rijndael-ammended.pdf
32. G. Tang, X. Liao, A method for designing dynamical s-boxes based on discretized chaotic map. Chaos Solitons Fract. **23**(5), 1901–1909 (2005)
33. M. Khan, T. Shah, H. Mahmood, M.A. Gondal, I. Hussain, A novel technique for the construction of strong s-boxes based on chaotic Lorenz systems. Nonlinear Dynam. **70**(3), 2303–2311 (2012)
34. A. Belazi, A.A.A. El-Latif, A.-V. Diaconu, R. Rhouma, S. Belghith, Chaos-based partial image encryption scheme based on linear fractional and lifting wavelet transforms. Opt. Lasers Eng. **88**, 37–50 (2017)

35. F. ul Islam, G. Liu, Designing s-box based on 4D-4wing hyperchaotic system. 3D Res. **8**(1), 1–9 (2017)
36. F. Özkaynak, Construction of robust substitution boxes based on chaotic systems. Neural Comput. Appl. 1–10 (2017)
37. I. Hussain, T. Shah, M.A. Gondal, Y. Wang, Analyses of SKIPJACK s-box. World Appl. Sci. J. **13**(11), 2385–2388 (2011)

Chapter 5
Modification of the Quantum Logistic Map with Application in Pseudo-Random Bit Generation and Image Encryption

Ioannis Kafetzis and Christos Volos

Abstract This work introduces a novel chaotic map defined by composition of the quantum logistic map and the sine map. The dynamic behavior of this map is thoroughly investigated and is shown to be more complex than that of the original quantum logistic map. Sequentially, a PRBG is defined using the values of the introduced map, the validity of which is determined via the NIST statistical suite. Finally, both the chaotic map and the PRBG are utilized in the definition of an encryption scheme for grayscale images. Results in cryptography are validated by testing.

Keywords Quantum chaos · Quantum logistic map · Image encryption · Secure communications

5.1 Introduction

The digitization of communications and modern technological advancements, such as IoT applications, has led to an enormous increase of the amount of data being transferred online. A big part of these data are images, ranging from photographs to medical images. A natural consequence is an ever-increasing need for securing digital communications against unauthorized third parties that may try to intercept or tamper the signal being transmitted. The scientific field that focuses on determining schemes that guarantee secure communications is cryptography, a branch of mathematics, physics, and computer science. The goal of cryptography is to transform given data into a form that does not allow information about the original data to be extracted. Furthermore, it is important that the original data can be retrieved from the cipher by the authorized receiver [1]. One problem is that in classical encryption schemes [2], for example the DES, tDES, etc., their

I. Kafetzis (✉) · C. Volos
Laboratory of Nonlinear Systems - Circuits and Complexity, Physics Department, Aristotle University of Thessaloniki, Thessaloniki, Greece
e-mail: kafetzis@physics.auth.gr; volos@physics.auth.gr

G. Huerta Cuellar et al. (eds.), *Complex Systems and Their Applications*,
https://doi.org/10.1007/978-3-031-02472-6_5

performance both runtime and randomness of result when applied in image file encryption is limited [3, 4].

This has led to the development of chaos-based encryption techniques for images, which perform well on both time efficiency and randomness of the resulting cipher image [5, 6]. Chaos theory is a well-established area of mathematics, which finds numerous applications in physics [7, 8]. Its main focus lies on chaotic systems, which are nonlinear dynamical systems, either in continuous or discrete time, which demonstrate extreme sensitivity to initial conditions [9, 10]. In other words, trajectories of a chaotic system that start from almost identical initial conditions diverge. Some of the most famous examples of such systems in continuous time include the Lorenz system [11, 12]. Examples of discrete time chaotic systems that also find applications in cryptography include the logistic map [13], sine map [14], and Henon map [15].

The phenomenal results of chaos-based encryption schemes have led researchers in defining novel chaotic systems that demonstrate more complex chaotic behavior, since this potentially leads to more secure encryption schemes [16–18]. Chaotic systems also constitute an excellent source of randomness, as they are widely used for defining Pseudo-Random Bit Generators (PRBGs for short) that are methods for generating random bits. A plethora of methodologies that allow the values of chaotic systems to be used for PRBGs is being proposed in recent years [19, 20] and still is a subject that gathers research interest. The randomness of such PRBGs is verified using a Statistical Test Suite proposed by the National Institute of Standards and Technology (NIST) [21]. This suite contains a total of fifteen (15) statistical tests, used to determine if a given bitstream is random or not, based on the computation of p-values. A bit generator can be considered to be random, if it successfully passes all of the tests included in the suite.

Most chaos-based image encryption schemes constitute two pieces, confusion and diffusion. Confusion is the process of altering the bits such that the cipher text does not reveal any correlation that might exist between the bits of the original image. This is usually achieved by using the exclusive OR (XOR for short) operation between the bits of the original image and random bits generated through a PRBG [22, 23]. Diffusion on the other hand is the process of making the cipher image sensitive to even small alterations of the original image. Two of the most common ways to achieve diffusion is the shuffling of pixels which leads to diminishing of any local correlation of pixel values and the definition of image encryption schemes where the keys are plaintext depended [24]. The latter is of vast significance, since even a slight alteration of the plaintext image leads to different initial conditions for the chaotic system that is used in the encryption scheme. This in turn results into the chaotic system having a completely different trajectory and hence the resulting cipher image differs significantly.

Quantum chaos is a new research field combining quantum mechanics, quantum computation, and chaos theory [25–27]. Examples of quantum chaotic maps are the quantum Baker map [28] and quantum cat maps [29]. Another well known discrete quantum chaotic map is the quantum logistic map (QLM for short), introduced in

[30], which has been utilized in the definition of numerous encryption and secure communication schemes [31–36]. This map, for real valued initial conditions, is described by the following system of difference equations:

$$\begin{aligned} x_{n+1} &= r(x_n - |x_n|^2) - r y_n \\ y_{n+1} &= -y_n e^{-2b} + e^{-b} r \left[(2 - 2x_n)\, y_n - 2x_n z_n\right] \\ z_{n+1} &= -z_n e^{2b} + e^{-b} r \left[2(1 - x_n) z_n - 2x_n y_n - x_n\right] \end{aligned} \tag{5.1}$$

The name is given due to the logistic map that appears in the equation of the first state. Although this map has been used for cryptography applications, it has two major weaknesses. The first one is that the system becomes unstable for values for the parameter r that are greater than 4, which is inherited by the classical logistic map. The second is that the system is chaotic for a limited amount of values of r.

In this work, a modification of the QLM is proposed, which overcomes these weaknesses while also enhancing the complexity of the behavior of the map. This is achieved by utilizing a logic similar to that of sine chaotification [37, 38], according to which, composing any discrete chaotic map with the sine map leads to a new chaotic map with larger Lyapunov exponent. Hence, the proposed system is defined by composing the logistic part of the QLM with the absolute value of the sine map. This map is stable for all values of parameter r. So, it is chaotic for a vast amount of values of its parameters and demonstrates more complex behavior than that of the original QLM. The proposed map is then utilized in the definition of a PRBG, the validity of which is verified using the NIST statistical suite. Finally, the proposed map and the PRBG are utilized in the definition of an encryption scheme for grayscale images. The design of the method is based on the idea of splitting the plaintext image into subimages, each of which is utilized for the encryption of the others [39–41]. This essentially guarantees that even a slight alteration of the plaintext image leads to a different cipher image. Examples where this method is applied are presented and the security of the method against some of the most common attacks is verified via statistical measures such as the histogram and correlation analysis, information entropy, and resistance to differential attacks.

The rest of the work is organized as follows. In Sect. 5.2 the quantum logistic map is analyzed and Sect. 5.3 presents the idea of sine chaotification, which is the basis for the proposed map. Subsequently, the proposed map is given in Sect. 5.4, together with its dynamic analysis, which is compared with that of the QLM. After that, Sect. 5.5 contains the definition of the PRBG and the results of the NIST test and in Sect. 5.6 the proposed image encryption scheme, examples of its applications, and the result of the test for its security against different types of attacks are presented. Finally, Sect. 5.7 concludes the work.

5.2 The Quantum Logistic Map

A quantum chaotic map that has been utilized in numerous applications is the quantum logistic map (or QLM), which is a three-dimensional quantum chaotic system introduced in [30], and whose states are in general complex numbers. Thus, in the general case the QLM is described by the following set of state equations:

$$\begin{aligned} x_{n+1} &= r(x_n - |x_n|^2) - r y_n \\ y_{n+1} &= -y_n e^{-2b} + e^{-b} r \left[(2 - x_n - \overline{x}_n)\, y_n - x_n \overline{z}_n - \overline{x}_n z_n\right] \\ z_{n+1} &= -z_n e^{2b} + e^{-b} r \left[2(1 - \overline{x}_n) z_n - 2 x_n y_n - x_n\right] \end{aligned} \tag{5.2}$$

where $\overline{c}$ denotes the conjugate of the complex number c. For its description, the system has three states, namely x, y, and z and two parameters, b and r.

If the initial conditions $(x_0,\ y_0,\ z_0)$ and the parameters b and r are all chosen to be real numbers, then $(x_k,\ y_k,\ z_k) \in \mathbb{R}^3$ for every point k of the trajectory. This case allows (5.2) to take a simpler form, by replacing the conjugate. Thus, for the case of real valued initial conditions and parameters, the system is described as seen next.

$$\begin{aligned} x_{n+1} &= r(x_n - |x_n|^2) - r y_n \\ y_{n+1} &= -y_n e^{-2b} + e^{-b} r \left[(2 - 2x_n)\, y_n - 2 x_n z_n\right] \\ z_{n+1} &= -z_n e^{2b} + e^{-b} r \left[2(1 - x_n) z_n - 2 x_n y_n - x_n\right] \end{aligned} \tag{5.3}$$

One weakness that the QLM has is that it is not chaotic for small values of r, and becomes unstable for values of r greater than 4. This is inherent from the classical logistic map.

Thus, in some occasions, when considering the map for cryptographic applications, the parameter r is considered to have a value close to 3.99 and b takes values larger than 6 [42–44].

Hence, our goal is to determine a new system that overcomes these weaknesses of the QLM. More explicitly, the introduced system shall have a larger Lyapunov exponent, overcoming the problem of instability for $r \geq 4$. Furthermore, the goal is to determine a map for which there are intervals (r_{min}, r_{max}) and (b_{min}, b_{max}) such that the system is chaotic for almost any selection of parameters $(r, b) \in (r_{min}, r_{max}) \times (b_{min}, b_{max})$.

5.3 Method for Achieving More Complex Chaotic Behavior

Chaotic systems are a constant point of interest in Pseudo-Random Bit Generation and in the definition of encryption schemes. The complexity of the system usually has great effect on designing a successful PRBG, something that has led to

the development of methods that lead to systems with more complex chaotic behavior. One of the most prominent methods that yield such results is that of sine chaotification, introduced in [37]. Put plainly, composing the iterative step of any chaotic system with the sine map, described as

$$x_{k+1} = m \sin(\pi x_k) \tag{5.4}$$

where m is a parameter, leads to a system with a more complex chaotic behavior.

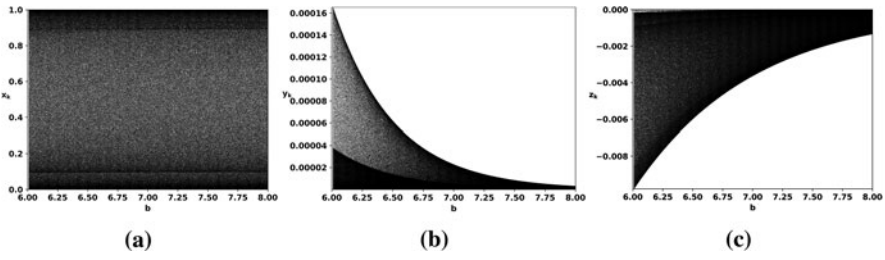
It shall be noted that this composition also removes any instability problems the original map may have, since the sine map guarantees that the system values lie inside the closed interval $[-1, 1]$. This idea shall form the basis for the modified version of the QLM that is proposed in this work.

5.4 The Sine-Quantum Logistic Map

The discussion of the previous section clarifies the idea behind the derivation of the proposed map, namely the *sine-quantum logistic map*. For the proposed map, the QLM is considered to have real valued initial conditions, and is thus described as in (5.3). For this system, its "logistic part," which is the $r \cdot (x_k - |x_k|^2)$ is composed of the absolute value of sine map $\sin(\pi x_k)$ for $m = 1$. The difference equations describing the resulting matrix are shown next.

$$\begin{aligned} x_{n+1} &= |\sin\left(\pi r(x_n - |x_n|^2)\right)| - r y_n \\ y_{n+1} &= -y_n e^{-2b} + e^{-b} r\left[(2 - 2x_n)\, y_n - 2x_n z_n\right] \\ z_{n+1} &= -z_n e^{2b} + e^{-b} r\left[2(1 - x_n) z_n - 2x_n y_n - x_n\right] \end{aligned} \tag{5.5}$$

The investigation of the behavior of the proposed map begins by considering the bifurcation diagram with respect to parameter b. The bifurcation diagrams for all three states x_k, y_k, and z_k of (5.5) are shown in Fig. 5.1.



(a) (b) (c)

Fig. 5.1 Bifurcation diagram of the states (**a**) x (**b**) y and (**c**) z of the system (5.5), *with respect to the parameter* b for initial conditions $(x_0, y_0, z_0) = (0.6, 0.05, 0.1)$ and $r = 3.99$

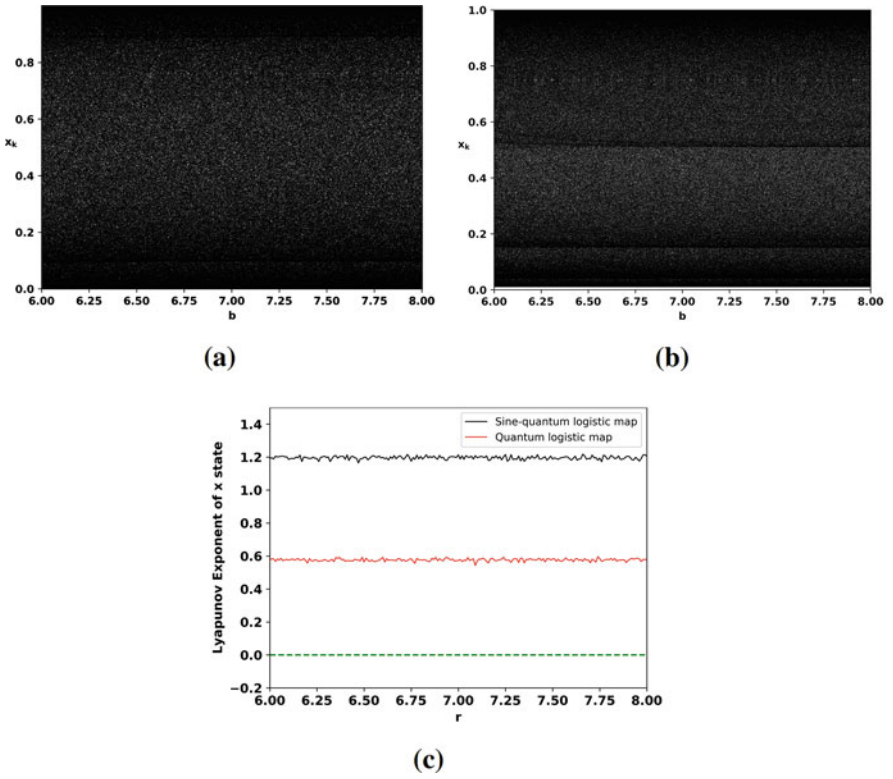


Fig. 5.2 (**a**) Bifurcation diagram of the x state of the system (5.5), (**b**) bifurcation diagram of the x state of the QLM, (**c**) Common Lyapunov exponent for the x state of both system (5.5) and the QLM *with respect to the parameter b* for initial conditions $(x_0, y_0, z_0) = (0.6, 0.05, 0.1)$ and $r = 3.99$

There are two important conclusions that can be drawn from Fig. 5.1. The first one is that the bifurcation diagram for the x state is dense, with values covering the $[0, 1]$ interval, which is desirable when considering cryptography related applications. On the other hand, it can be seen in subfigures (b) and (c) of Fig. 5.1 that the value range of the y and z states is limited and constantly decreases as the values of b increase. For this reason, for the rest of this work, only the x state is considered.

Moving on, the behavior of the x state for the proposed map (5.5) and the QLM are compared. In many works, the QLM is utilized by taking $r = 3.99$ and using b as a varying parameter. Thus, the first comparison for the systems takes this case into consideration. Indeed, the bifurcation diagram of the proposed system (a), the bifurcation diagram of the QLM (b), and the Lyapunov exponents for both systems (c) is presented in Fig. 5.2. At this point it should be mentioned that the computation of the Lyapunov exponent is performed using the QR factorization method from [45].

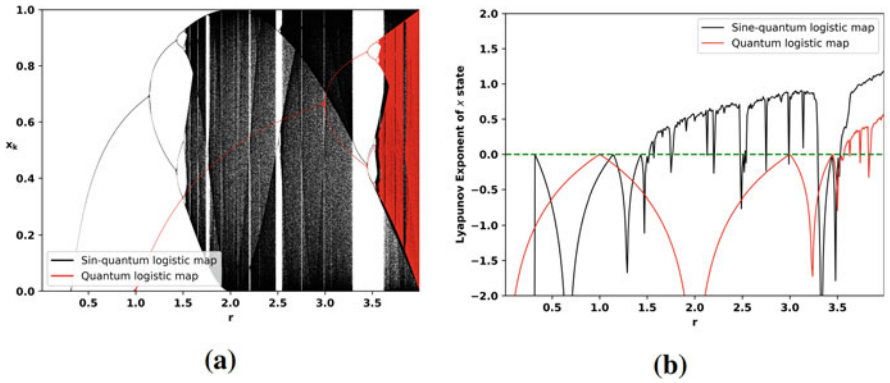


Fig. 5.3 (**a**) Bifurcation diagrams of the x state for the system (5.5) and the QLM (**b**) Lyapunov exponents of the x state of the system (5.5) and the QLM *with respect to the parameter* r for initial conditions $(x_0, y_0, z_0) = (0.6, 0.05, 0.1)$ and $b = 7$

It is seen in Fig. 5.2 that the proposed map has a much more complex behavior in contrast with the QLM. This can be verified by the difference in their Lyapunov exponents, which indicates that two trajectories starting from almost identical initial conditions diverge faster when considering the system (5.5).

Moving on, focus is turned to the behavior of the proposed system (5.5). This is performed in two steps. Initially, the behavior for $0 < r < 4$ is examined. This allows comparing the behavior of the proposed system with that of the QLM, since the QLM becomes unstable for $r > 4$. In Fig. 5.3, the bifurcation diagrams and Lyapunov exponents for the x states of both systems with respect to the parameter r are depicted. It can be easily verified that the proposed map has much more complex behavior than that of the original map.

The next step is to examine the behavior of the proposed map for the case where $r > 4$. The bifurcation diagram and the diagram of Lyapunov exponent for the x state of the system are shown in Fig. 5.4. An important observation is that intervals for the parameter r can be found, for example for $r \in (6, 7)$ such that the system is chaotic for almost every value of r inside such intervals.

As is to be expected, the system is stable and is in chaotic state for several values of r.

Closing the section of the dynamical analysis the return map and cobweb diagrams of the proposed map (5.5) and the QLM are studied. The return diagram of the proposed map and the QLM is depicted in Fig. 5.5. Although both return diagrams have a distinguishable structure, the return map of the proposed map is more complex. The significance of this complexity difference of the return maps can be seen through the Cobweb diagrams of the two systems, which are shown in Fig. 5.6

The dynamical analysis of the proposed map along with the extensive comparison with the QLM yield that the proposed system has a complex chaotic behavior. This fact renders the system a noteworthy option to use in the definition of PRBGs and encryption schemes.

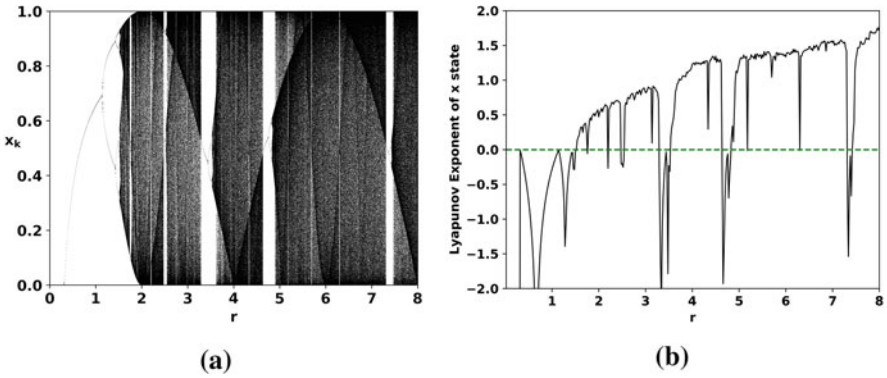


Fig. 5.4 (**a**) Bifurcation diagram of the x state of the system (5.5), (**b**) Lyapunov exponent diagram of the x state of (5.5) *with respect to the parameter* r for initial conditions $(x_0, y_0, z_0) = (0.6, 0.05, 0.1)$ and $b = 7$

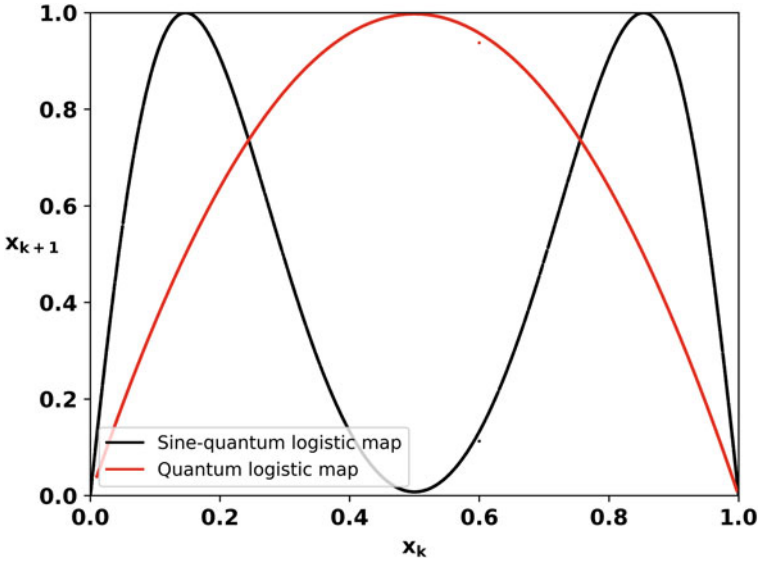


Fig. 5.5 Common diagram of the return maps of the proposed system and the QLM

5.5 Pseudo-Random Bit Generation

Considering that the x state of the proposed map (5.5) is chaotic with complex behavior makes it likely that it can be used for the definition of a PRBG. Indeed, this section is dedicated into providing a method for generating random bits based on (5.5) and verifying the validity of the PRBG via the results of the NIST test suite. The steps for determining a random bit are as follows. For any point of the trajectory

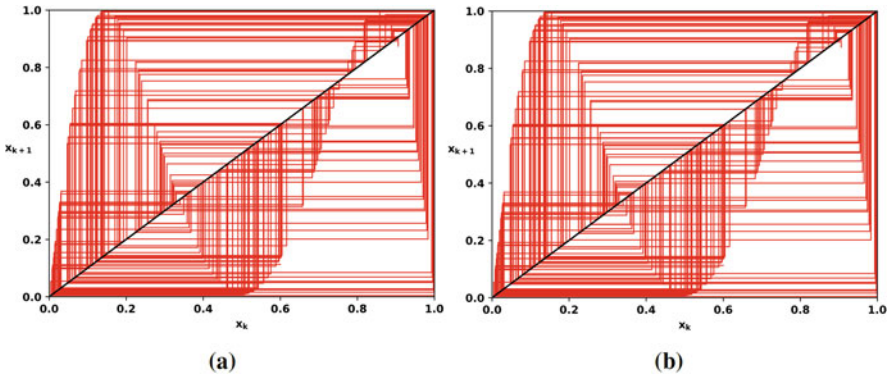


Fig. 5.6 Cobweb diagrams for (**a**) the x state of the system (5.5) and (**b**) the x state of the QLM for initial conditions $(x_0, y_0, z_0) = (0.6, 0.05, 0.1)$ and parameter values $r = 6$ and $b = 7$

(x_k, y_k, z_k) of the trajectory, the value $10^{10}x_k$ is calculated and then the modulo 2 operator is applied to it. This results in a value that lies in the interval $[0, 2)$. Finally, the bit value is obtained by comparing the latter with 1. The mathematical formulation for the PRBG is given next.

$$\{0, 1\} \ni \beta = \begin{cases} 0, & \text{if} \quad \text{mod}\ (10^{10} \cdot x_k, 2) > 1 \\ 1, & \text{if} \quad \text{mod}\ (10^{10} \cdot x_k, 2) \leq 1 \end{cases} \tag{5.6}$$

The validity of the PRBG is verified using the statistical suite provided by NIST, using a set of 100 bitstreams, each consisting of 10^6 bits. The statistical suite constitutes 15 randomness tests, each of which returns a p-value. A test is considered successful, if the returned p-value is greater than a threshold, which in our case has the default value of 0.01 and a generator is considered random if it successfully passes all of the tests. The results for the proposed generator, for initial conditions $(x_k, y_k, z_k) = (0.6, 0.05, 0.01)$ and parameters $(\beta, r) = (7, 6)$ are shown in Table 5.1. It should be noted that in the cases where multiple results were returned for the same test, for example the Nonoverlapping Template, the generator was successful in all iterations and the value presented in Table 5.1 is that of the last yielded result.

This PRBG is utilized in the next section to define a symmetric encryption scheme for monochromatic image.

5.6 Image Encryption Scheme

In this section, both the proposed map (5.5) and the PRBG introduced in Sect. 5.5 are utilized to define an image encryption scheme for grayscale images. Any such image is represented as a two-dimensional matrix whose entries, known as pixels,

Table 5.1 Results of the NIST suite tests for initial conditions $(x_0, y_0, z_0) = (0.6, 0.05, 0.01)$ and parameter values $(b, r) = (7, 6)$

No.	Test	Chi-square p-value	Score
1	Frequency	0.554420	98/100
2	Block frequency	0.048716	99/100
3	Cumulative sums	0.779188	99/100
4	Runs	0.779188	98/100
5	Longest run	0.085587	98/100
6	Rank	0.366918	100/100
7	FFT	0.275709	99/100
8	Nonoverlapping template	0.996335	100/100
9	Overlapping template	0.181557	97/100
10	Universal	0.514124	100/100
11	Approximate entropy	0.383827	98/100
12	Random excursions	0.452799	65/65
13	Random excursions variant	0.517442	65/65
14	Serial	0.401199	100/100
15	Linear complexity	0.554420	98/100

are integers with values ranging from 0 to 255. For the rest of the work, an image and its matrix representation are considered the same thing. In this context, phrases like "the rows/columns of an image" make sense, since it actually refers to the rows/columns of its matrix representation.

5.6.1 Outline of the Proposed Method

Next, the outline of the encryption method is presented in Algorithm 1. Following this, the technicalities of each step are presented separately, for the sake of clarity. Finally, the step to revert each step of the encryption method, that is, the decryption steps are presented in Algorithm 2.

What is worth noting is the dependencies between the encrypted blocks. The encryption (as well as the decryption) keys for block number k, for $k > 1$ the encrypted blocks $1, \ldots, k-1$ are required. On the other hand, determining the encryption (and decryption) key for the first block requires blocks number 2 to 9 having the same values after performing the original shift. Hence, in the decryption process, correctly decrypting the first block can only be achieved if all of the previous blocks have been decrypted correctly.

As for the decryption process, the images are just taken in reverse order.

Considering that the outline for the encryption and decryption steps has been clarified, the specific methods utilized in the different steps shall now be investigated.

Algorithm 1 Outline of the image encryption process

Require: Original Image
1: Initialize an empty list ℓ
2: Get a key value for the original image
3: Get the row and column shuffle indices for the original image
4: Shuffle the rows and then the columns
5: Break the image into 9 block sub images
6: Use blocks $2-9$ to obtain a key
7: Use the key from the previous step to encrypt block 1
8: Add the result of the encryption to the list ℓ
9: **for** $k = 2:9$ **do**
10: Use the encrypted blocks in list ℓ to determine a key
11: Use the key to encrypt the $k-$th block
12: Add the encrypted block to the list ℓ
13: **end for**
14: Put the blocks in place to form the encrypted image

Algorithm 2 Outline of the image decryption process

Require: Encrypted image
1: Initialize an empty list d
2: Split the encrypted image into 9 parts
3: **for** $k = 9:2$ **do**
4: Use blocks 1 to k to obtain a key
5: Decrypt the $k-$th block using the previous key
6: Add the decrypted block to the list d
7: **end for**
8: Use the block images contained in list d to decrypt block 1
9: Place the blocks in order to create the shuffled image
10: Use the image from the previous step to obtain a key
11: Use the key to produce the row and column shuffle indices
12: Revert the shuffling of the columns and then the rows to obtain the decrypted image

5.6.2 Determination of Shuffling/Encryption Key

Consider a non-empty list of images $\mathrm{Im}_1, \ldots, \mathrm{Im}_k$. The pixel values and locations of each image are utilized in determining the key, that is, the initial condition $(x_0,\ y_0,\ z_0)$ and parameter values b and r for the system. The steps and mathematical formulas utilized to determine the keys are presented in Algorithm 3. It should be observed that in this process, the pixel values are taken into consideration, thus granting keys that are plaintext dependent. What is not taken into consideration is the exact position in which values appear. This implies that if we consider an image Im_1 and let Im_2 be the result of shuffling the rows and columns of Im_1, then the key determined using Im_1 and Im_2 coincides.

Clearly, this process can also be performed when the input is a single image.

Algorithm 3 Determination of encryption/decryption key via block images

Require: List of block images $[\mathrm{Im}_1, \ldots, \mathrm{Im}_k]$, parameter q

Initialize $\widehat{x}, \widehat{y}, \widehat{z}, \widehat{b}, \widehat{r} = 0, 0, 0, 0, 0$

for i = 1 : k **do**

  Get the dimensions of Im_k, namely a_k, b_k

  $m_k = mean(\mathrm{Im}_k)$, $sv_k = sum(\mathrm{Im}_k)$, $sd_k = a_k + b_k$, $pd_k = a_k \cdot b_k$

  $\widehat{x} = \widehat{x} + {}^{sv_k + sd_k}/_{pd_k} \cdot q$

  $\widehat{y} = \widehat{y} + pd_k \cdot m_k + q_k \cdot sd_k$

  $\widehat{z} = \widehat{z} + m_k \cdot q_k + sd_k \cdot pd_k$

  $\widehat{b} = \widehat{b} + q_k \cdot \begin{cases} {}^{(i+j)}/_{sd_k}, & \text{if } i + j > m_k \\ {}^{(i+j)}/_{pd_k}, & \text{otherwise} \end{cases}$

  $\widehat{r} = \widehat{r} + q \cdot \begin{cases} {}^{\sqrt{i+j}}/_{sd_k}, & \text{if } i + j > m_k \\ {}^{(i-j)^2}/_{pd_k}, & \text{otherwise} \end{cases}$

end for

// Calculate the values for the key

$x_0 = \mod(\widehat{x}, 1) \in (0, 1)$

$y_0 = \dfrac{\mod(\widehat{y}, 9) + 1}{100} \in (0.01, 0.1)$

$z_0 = \dfrac{\mod(\widehat{z}, 9) + 1}{100} \in (0.01, 0.1)$

$b = 6 + \mod(\widehat{b}, 2) \in (6, 8)$

$r = 7 - \mod(\widehat{r}, 1) \in (6, 7)$

5.6.3 Row/Column Shuffling Process

It is assumed that a key for the system (5.5) has been determined and that the image to be shuffled has m rows and n columns. Then the shuffle indices for the rows are obtained as shown in Algorithm 4.

For the encryption process, the rows are shuffled first followed by the columns. For the shuffling process, the list of integers $\mathcal{I}_r$ and $\mathcal{I}_c$ obtained via Algorithm 4 define permutations for the row and column indices, respectively.

A very important observation is that the computation of the shuffle indices is independent of the pixel values of the image and only depend on the dimensions of the image being shuffled. Hence, using the same key leads to the same shuffle images for all images that have the same dimensions.

For the un-shuffling process the sets $\mathcal{I}_r$ and $\mathcal{I}_c$ are determined according to Algorithm 4, and the inverse permutation matrices are computed. Thus, the un-shuffling of the columns is performed first, followed by that of the rows.

5.6.4 Image Splitting

Splitting an image into parts is achieved by considering specific "cut" locations based on the dimensions of the image. For this, let m be a positive integer and define m_1, m_2, and m_3 as

Algorithm 4 Determination of the shuffle indices

Require: Number of rows m, number of columns n, system key
Initialize empty lists $\mathcal{I}_r = []$ and $\mathcal{I}_c = []$
while length$(\mathcal{I}_r) < m$ **do**
    Take the point x_k
    Calculate $w =$ mod $(10^{10} \cdot x_k), m)$
    if $w \notin \mathcal{I}_r$ **then**
        Add w as the last element of $\mathcal{I}_r$
    else
        Scrap w
    end if
    $k = k + 1$
end while
// At this point, the list for the shuffle indices for the rows have been determined and the determination of the column shuffle indices begins
while length$(\mathcal{I}_c) < n$) **do**
    Take the point x_k
    Calculate $w =$ mod $(10^{10} \cdot x_k), n)$
    if $w \notin \mathcal{I}_c$ **then**
        Add w as the last element of $\mathcal{I}_c$
    else
        Scrap w
    end if
    $k = k + 1$
end while

$$m_1 = \text{floor}\,(m/3)\,,\ m_2 = m_1 + \text{floor}\,((m-m_1)/3)\,,\ m_3 = m$$

One can observe that the relation $m_1 < m_2 < m$. Applying the above when m is the number of rows or columns of an image yields the sizes of the blocks.

5.6.5 Block Encryption/Decryption

In this subsection, the encryption and decryption of an image block are described, using the sub-processes discussed so far. Let I be an image block to be encrypted and let $\ell = [\text{Im}_1, \ldots, \text{Im}_k]$ be the non-empty list of image blocks that are to be used in the encryption of I. Initially the key that is to be used for the encryption of I is determined via the blocks contained in the list ℓ according to Algorithm 3. Having determined the key, the map is iterated so that the row and column shuffle indices $\mathcal{I}_r$ and $\mathcal{I}_c$ are computed as shown in Algorithm 4. Using these, the rows and subsequently the columns of the block are shuffled. Following this, the chaotic map is now used for the PRBG, so that 8 random bits, that is, a random byte, are produced for each pixel of the block image. Each pixel is first converted to its binary form and subsequently XORed with the corresponding random byte. The resulting binary representation is converted to integer and is used as the encrypted pixel value.

Performing this process for all pixels of the block, results into the encrypted block. The description of the algorithm is also given in Algorithm 5. For the decryption process, similar steps are followed, with key difference being the order in which the row and column shuffling is reversed. Initially, the key for the encryption of the block is determined via the designated block subimages. Observe that the subimages that take part in the calculation of the decryption key are exactly the same as the images used for obtaining the encryption key. Hence, the encryption and decryption keys coincide. The fact that both keys being identical combined with the fact that the dimensions of the original and encrypted blocks are the same, result into the exact same row and column shuffle indices that are actually obtained after the same number of system iterations. This implies that the shuffle indices for the encrypted block shall be determined first, but not used immediately. After computing the shuffle indices, the random bytes for each pixel can be determined, since they are to be used for decrypting the pixel as well. The process is based on a fundamental property of the XOR gate, here denoted by $\otimes$, which is $(a \otimes b) \otimes b = a$. Thus, the XOR operation between the binary form of the encrypted pixel and the random byte used in the encryption process results into the original value of each pixel. After performing this process for each pixel, the shuffling process is reverted. For this, the inverse permutation for the row and column indices can be determined. Finally, the columns are put into their original positions first, followed by the rows of the matrix. The steps for the decryption method are also summarized in Algorithm 6.

Algorithm 5 Encryption of each block

Require: Image block I, list of image blocks $\ell = [\mathrm{Im}_1, \ldots, \mathrm{Im}_k]$
Calculate the key for the chaotic system via ℓ
Get the dimensions of I $m, n = size(I)$.
Get the row and column shuffle indices $\mathcal{I}_m$ and $\mathcal{I}_n$
Shuffle the rows of I
Shuffle the columns
for $i = 1 : m$ **do**
    for $j = 1 : m$ **do**
        Determine 8 random bits using the PRBG proposed in Sect. 5.5
        Form a random byte $b_{i,j}$
        Convert the pixel in position i, j to its binary form, namely $p_{i,j}$
        Set $I[i, j] = \mathrm{XOR}(p_{i,j}, b_{i,j})$
    end for
end for
return Encrypted block

5.6.6 *Security Analysis of the Proposed Method*

The remaining of this section is dedicated to studying the security of the proposed method. The results of the method are presented via examples where the method

Algorithm 6 Decryption of each block

Require: Encrypted image block I, list of image blocks $\ell = [\text{Im}_1, \ldots, \text{Im}_k]$
Calculate the key for the chaotic system via ℓ
▷Get the dimensions of I $m, n = size(I)$.
Get the row and column shuffle indices $\mathcal{I}_m$ and $\mathcal{I}_n$
for $i = 1 : m$ **do**
  for $j = 1 : m$ **do**
    Determine 8 random bits using the PRBG proposed in Sect. 5.5
    Form a random byte $b_{i,j}$
    Convert the pixel in position i, j to its binary form, namely $p_{i,j}$
    Set $I[i, j] = \text{XOR}(p_{i,j}, b_{i,j})$
  end for
end for
Find the inverse permutation for the rows $\mathcal{I}_m^{-1}$ and for the columns $\mathcal{I}_n^{-1}$
Rearrange the columns of the block using $\mathcal{I}_n^{-1}$
Rearrange the columns of the block using $\mathcal{I}_m^{-1}$
return Decrypted block

is applied. The results of the encryption process on different images, taken from USC-SIPI Image Database (http://sipi.usc.edu/database/) are shown in Figs. 5.7 and 5.8. The security of the method is studied via statistical tests that demonstrate its resistance against common types of attacks.

Histogram One of the most fundamental security tests for image encryption method stems from the histogram of an image, which represents the distribution of different pixel values. An image containing some sort of information not all color intensities appear with the same frequency, which implies that the histogram of such images is usually not uniformly distributed. As a result the histogram can be utilized to derive information on whether an image contains information or not, without actually allowing any deductions for the content of the image. Hence, the goal of an image encryption scheme is to yield a histogram with distribution that is close to uniform, thus masking the fact that the encrypted image contains information. The histograms for both plaintext and encrypted image for the examples in study are shown in Figs. 5.9 and 5.10.

Correlation Analysis When considering an image, the value of a pixel is highly correlated with the values of adjacent pixels [46, 47]. This is to be expected, since such pixels tend to have similar color intensity, that is, similar values. The horizontal and vertical correlation, in other words the correlation between consecutive rows and columns of an image is a common statistical tool utilized in the literature for testing image encryption methods. The process for such test constitutes creating two images, one containing the rows (respectively columns) from first to pre-last and one containing the rows (resp. columns), converting those matrices into one dimensional vectors and computing their correlation.

In this work, a slight modification of this method is proposed. The correlation between consecutive rows and columns is given in the form of a diagram, where the

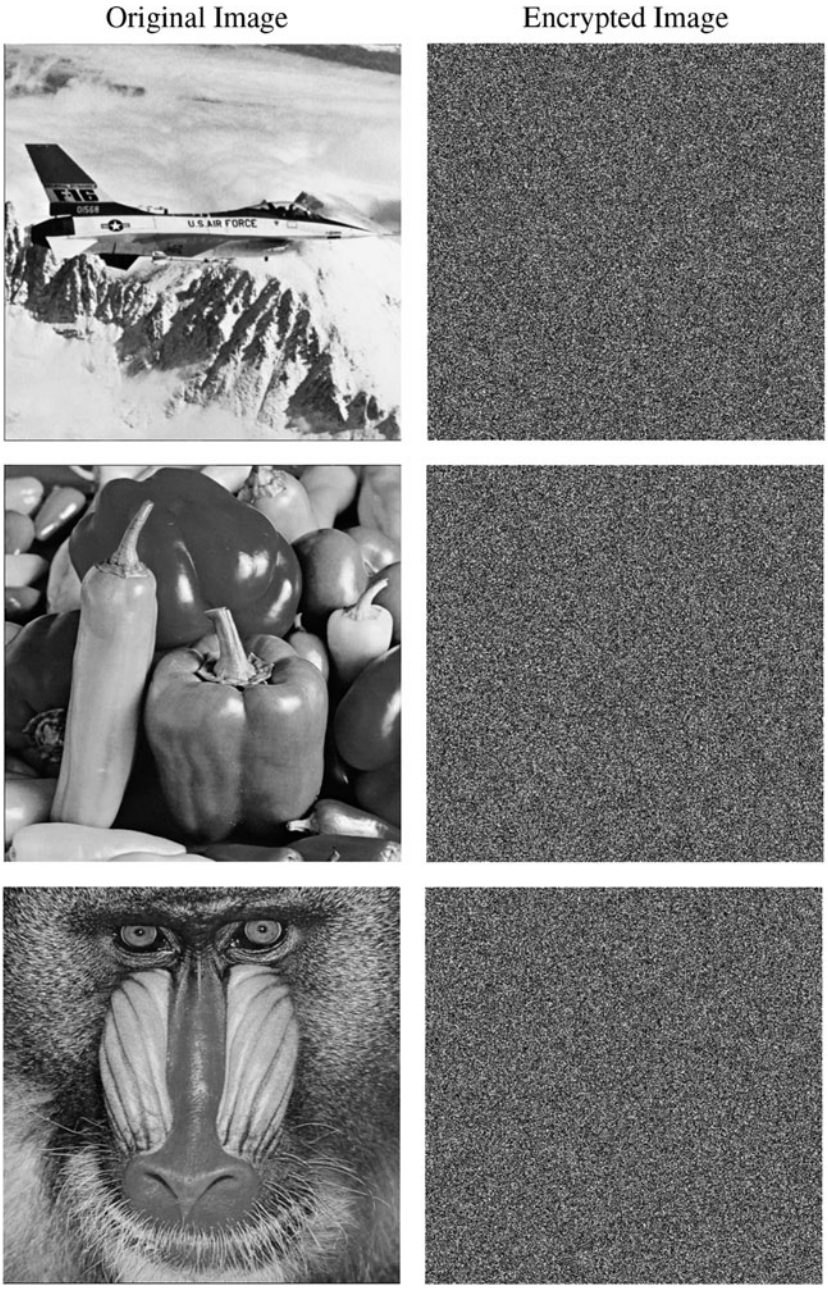


Fig. 5.7 Plaintext images and encrypted images after using the proposed method

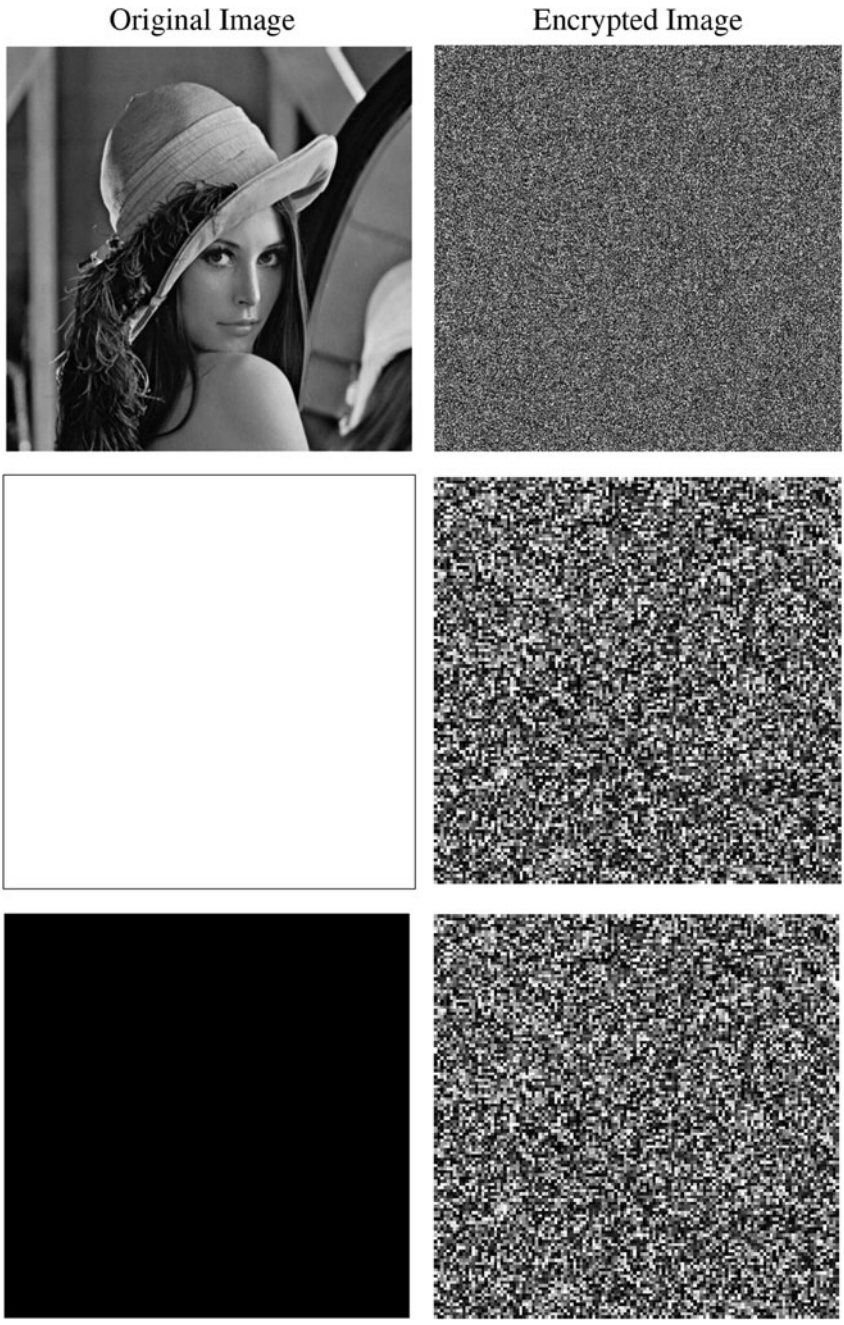


Fig. 5.8 Plaintext images and encrypted images after using the proposed method

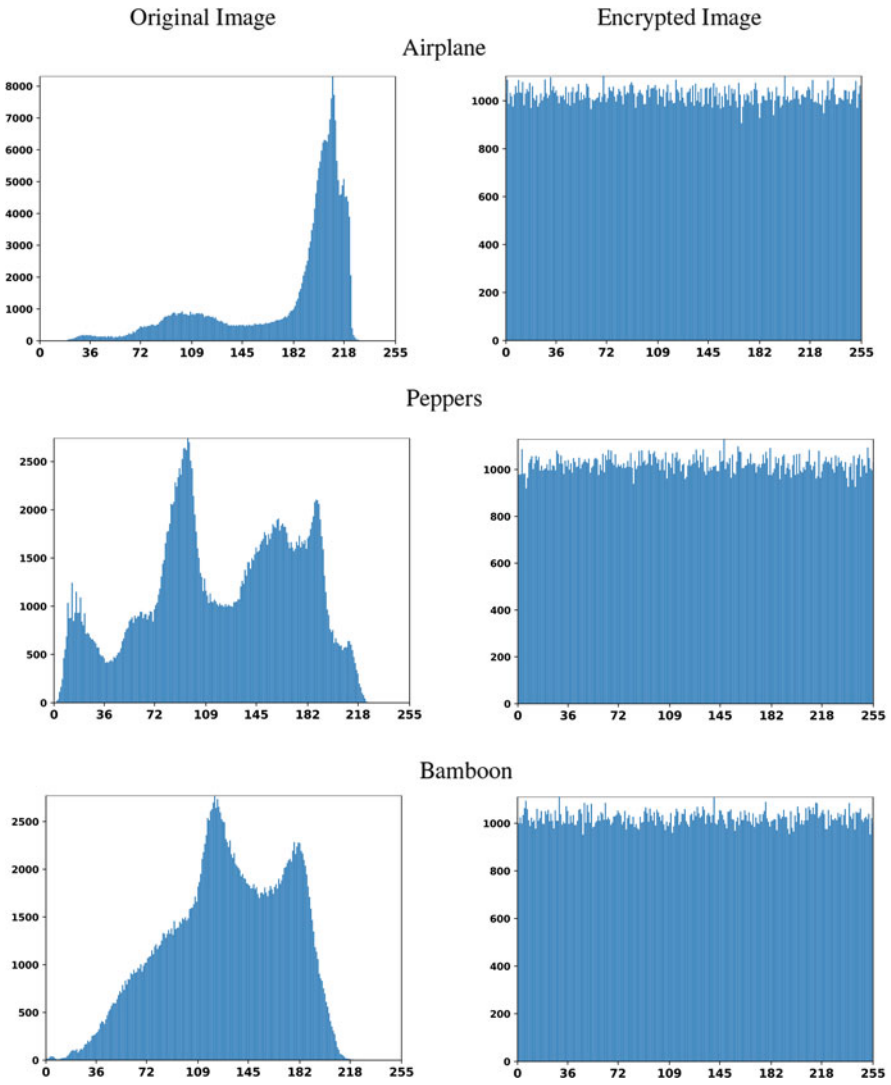


Fig. 5.9 (Left) Histogram of the original image (Right) Histogram of the encrypted image. The corresponding image name is shown above the diagrams

x represents the line number, ranging from the first until the pre-last column, and the y axis represents the correlation of the row or column with the succeeding one. It can be seen in Figs. 5.11 and 5.12 that this diagram has a structure for the original image. On the other hand, after the proposed encryption method is applied, the respective diagram for the original image has no structure and is closed to randomly scattered points.

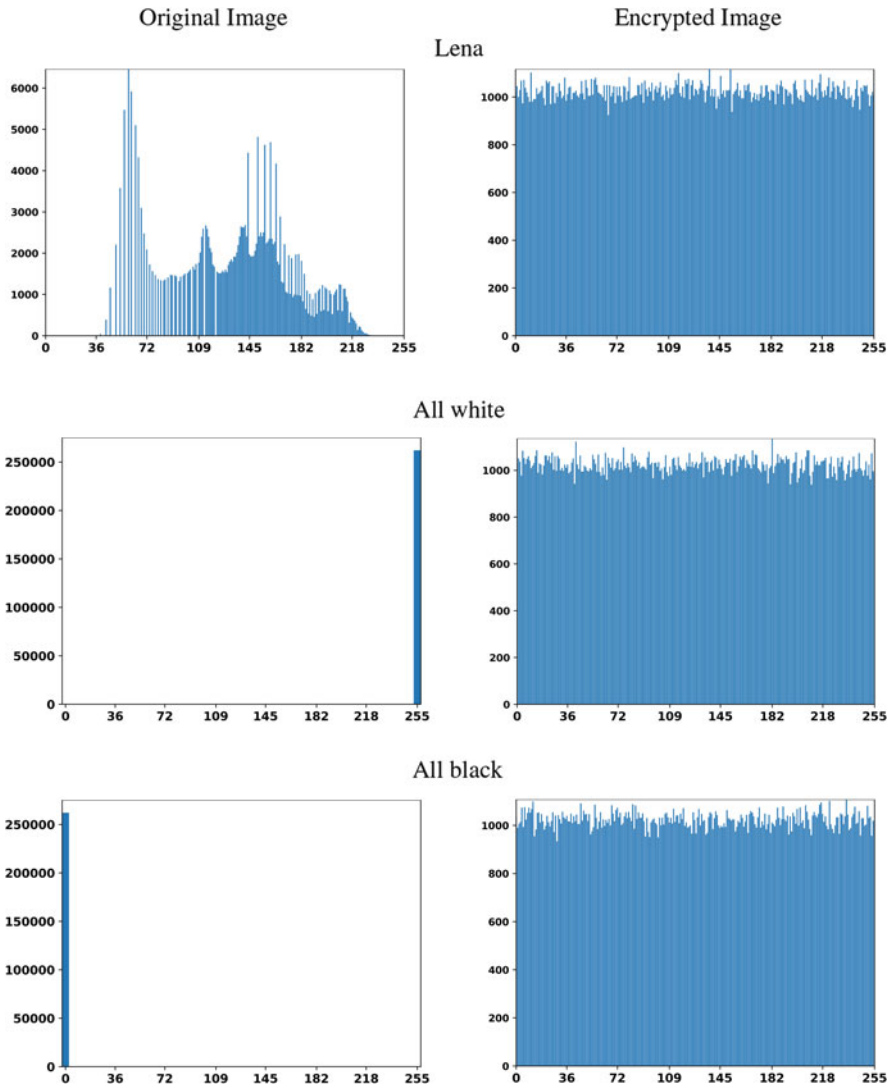


Fig. 5.10 (Left) Histogram of the original image (Right) Histogram of the encrypted image. The corresponding image name is shown above the diagrams

Information Entropy Information entropy is used as a measure of randomness for a signal [48–50]. When it comes to grayscale images, information entropy is computed as

$$H(S) = -\sum_{i=0}^{2^8-1} p(s_i)\log_2(p(s_i)) \tag{5.7}$$

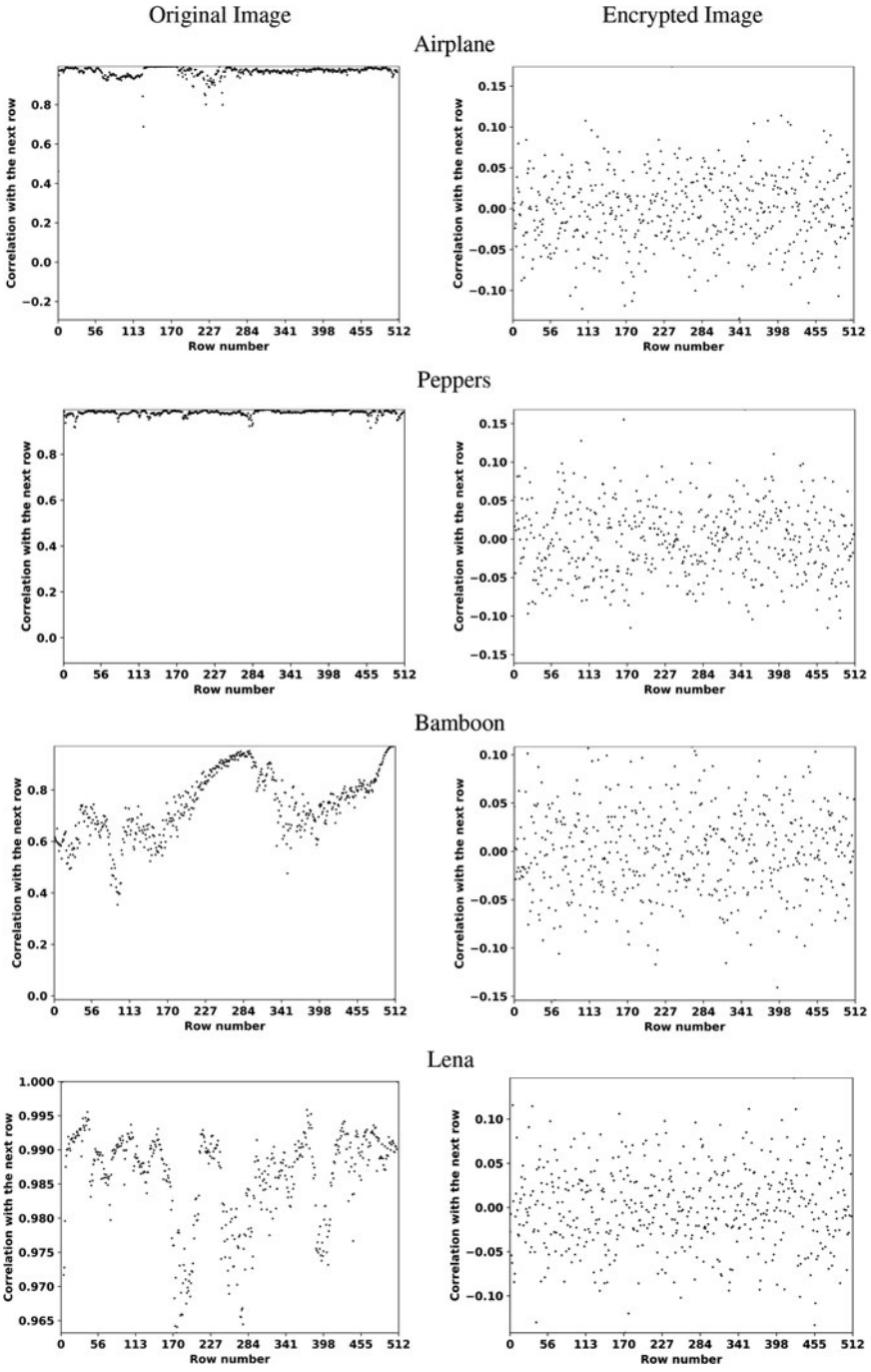


Fig. 5.11 Correlation of subsequent columns of the original and encrypted images

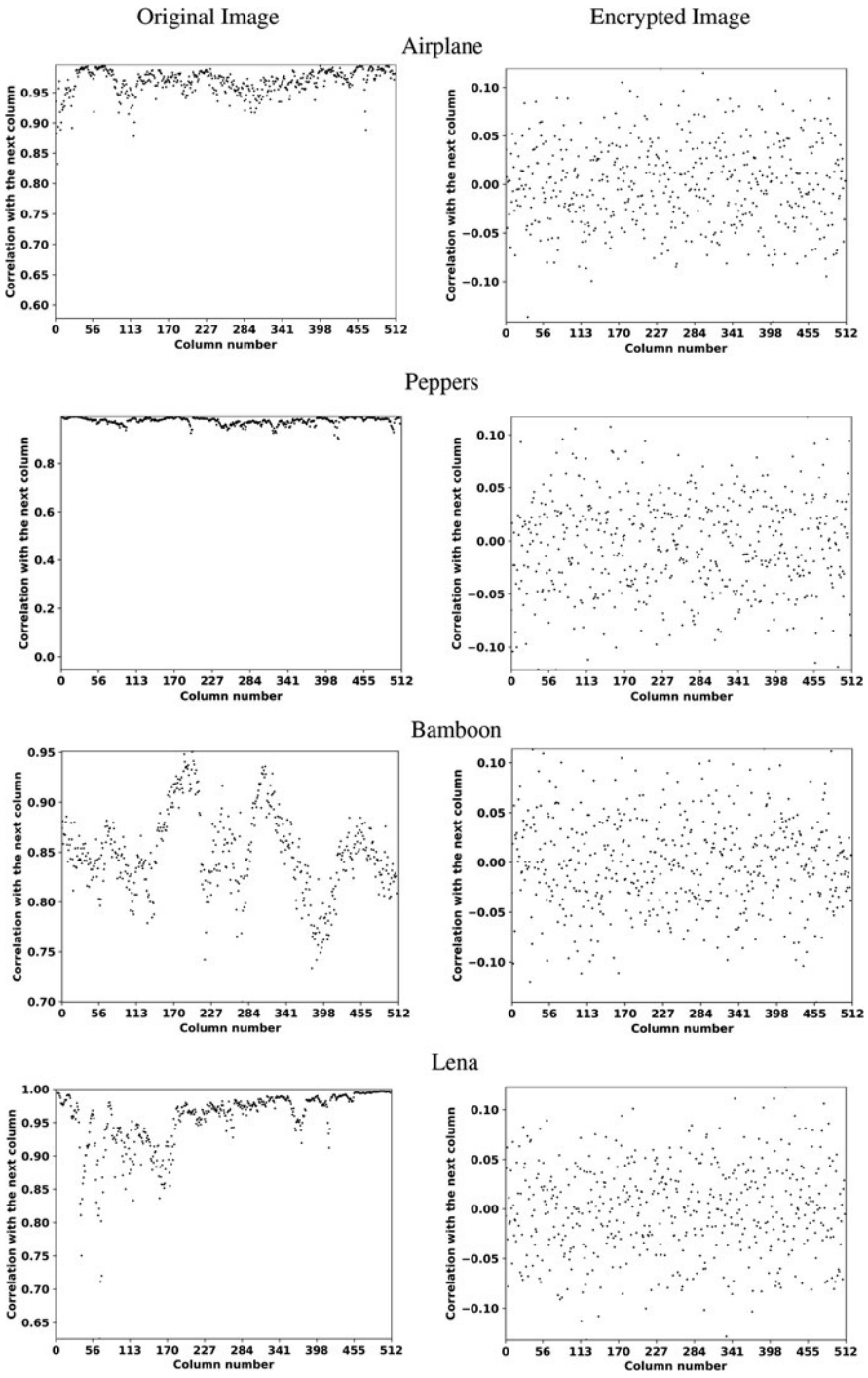


Fig. 5.12 Correlation of subsequent columns of the original and encrypted images

Table 5.2 Information Entropy of the original and encrypted images

Plaintext image	Original image	Encrypted image
Airplane	6.7024	7.9992
Peppers	7.5936	7.9993
Bamboon	7.3583	7.9994
Lena	6.9698	7.9993
All white	0	7.9880
All black	0	7.9882

where s_i is a possible value, in our case an integer from 0 to 255 that appears in the image and $p(s_i)$ is the probability with which s_i appears. For a grayscale image, a value close to 8 indicates that the pixel values of the image are random, thus characterizing the encryption method as resistant against entropy attacks. The values of the entropy for the original and encrypted images are shown in Table 5.2, from which it can be verified that the desired results are achieved.

Differential Analysis Attack An attack that is commonly utilized in order to extract information about the structure of the proposed method is constantly feeding the encryption algorithm with almost identical images, which is images that differ by one pixel. To ensure that an encryption scheme is secure against such attacks, the method has to guarantee that when applied to two plaintext images that differ by only one pixel, the resulting cipher images differ significantly.

There are two main measures used to quantify how two encrypted images obtained in such a context differ, namely the Number of Pixels Change Rate, or NPCR for short, and the Unified Average Changing Intensity (UACI) [51, 52]. These measures, when considering two such encrypted images I and $\widetilde{I}$ of dimensions $m \times n$, as computed as follows:

$$NPCR = \frac{\sum_{i=0}^{m}\sum_{j=0}^{n} d_{ij}}{m \cdot n} 100\% \tag{5.8}$$

$$UACI = \frac{\sum_{i=0}^{m}\sum_{j=0}^{n} |I_{ij} - \widetilde{I}_{ij}|}{255 \cdot m \cdot n} 100\% \tag{5.9}$$

where I_{ij} denotes the value of the pixel of image I in row i and column j, and d_{ij} is defined as

$$d_{ij} = \begin{cases} 0, & \text{if } I_{ij} = \widetilde{I}_{ij} \\ 1, & \text{otherwise} \end{cases}$$

Table 5.3 Results of the NPCR and UACI tests for different example images

Plaintext image	NPCR	UACI
Airplane	88.6528%	44.5541%
Peppers	88.6497%	44.4327%
Bamboon	88.6505%	44.5265%
Lena	88.6325%	44.4081%
All white	99.6704%	50.0920%
All black	99.6459%	49.709%

The fact that the encryption keys for each block depend on the values of the pixels guarantees that a change in the value of one pixel leads to a completely different encrypted image. The measures for the examples in study are presented in Table 5.3.

Key Space An image encryption scheme has to be secure against brute force attacks. This can be achieved by guaranteeing that the key space is greater than 2^{100}, as discussed in [53]. The key for the proposed map constitutes five values, the initial conditions (x_0, y_0, z_0) and the parameters b and r. Hence assuming 16 bit accuracy, a lower bound for such key is computed as follows:

$$10^{5\cdot 16} > 10^{4\cdot 20} = (10^4)^{20} >> (2^{10})^{20} = 2^{200} > 2^{100}$$

Hence the proposed encryption scheme is secure against attacks. It is further worth mentioning that different keys are computed throughout the encryption process, leading to an even greater key space.

5.7 Conclusions

In this work, a modification of the quantum logistic map has been proposed. The modified map has several advantages when compared to the QLM since it is stable for all values of the parameter r, is chaotic for a great amount of values of the parameter r and demonstrates much more complex behavior. This map has been utilized in the definition of a PRBG that was verified via the test of the NIST statistical suite. Finally, the proposed map and the PRBG are utilized in the definition of an encryption scheme for grayscale images. Several examples where the method is applied are given, and its security against attacks is verified through statistical measures.

This work can be a stepping stone for further research in several ways. Similar ideas for the development of chaotic maps with more complex behavior can be implemented. Furthermore, the proposed system can be utilized for the design of more PRBGs, which could potentially include the states y and z as well. Finally, the proposed encryption method could be altered so that it can be utilized for encryption of other signal types, such as images with color and video.

Acknowledgments The authors would like to thank the anonymous reviewers for their thorough inspection of the work and their insightful comments that helped improve the work.

References

1. M.Y. Rhee, *Internet Security: Cryptographic Principles, Algorithms, and Protocols* (Wiley, Hoboken, 2003)
2. M. John Justin, S. Manimurugan, A survey on various encryption techniques. Int. J. Soft Comput. Eng. **2**(1), 429–432 (2013). ISSN 2231:2307
3. S.F. El-Zoghdy, Y.A. Nada, A.A. Abdo, How good is the DES algorithm in image ciphering. Int. J. Adv. Netw. Appl. **2**(5), 796–803 (2011)
4. C.J. Mitchell, On the security of 2-key triple DES, in *On the Security of 2-Key Triple DES*. Royal Holloway, University of London, Information Security Group (2016)
5. M. Kumari, S. Gupta, P. Sardana, A survey of image encryption algorithms. 3D Res. **8**(4), 37 (2017)
6. S. Fadhel, M. Shafry, O. Farook, Chaos image encryption methods: asurvey study. Bullet. Electr. Eng. Inf. **6**(1), 99–104 (2017)
7. K. Kaneko, I. Tsuda, *Complex Systems: Chaos and Beyond: Chaos and Beyond: A Constructive Approach with Applications in Life Sciences* (Springer, Berlin, 2001)
8. G. Grassi, Chaos in the real world: recent applications to communications, computing, distributed sensing, robotic motion, bio-impedance modelling and encryption systems. Symmetry **13**(11), 2151 (2021)
9. S. Shaukat, A.L.I. Arshid, A. Eleyan, S.A. Shah, J. Ahmad, Chaos theory and its application: an essential framework for image encryption. Chaos Theory Appl. **2**(1), 17–22 (2020)
10. R.P. Murphy, *Chaos Theory* (Ludwig von Mises Institute, Auburn, 2010)
11. K. Sun, J.C. Sprott, Dynamics of a simplified Lorenz system. Int. J. Bifurcation Chaos **19**(4), 1357–1366 (2009)
12. A. Ouannas, A.T. Azar, S. Vaidyanathan, On a simple approach for QS synchronisation of chaotic dynamical systems in continuous-time. Int. J. Comput. Sci. Math. **8**(1), 20–27 (2017)
13. M. Ausloos, *The Logistic Map and the Route to Chaos: From the Beginnings to Modern Applications* (Springer, Berlin, 2006)
14. A. Belazi, A.A. Abd El-Latif, A simple yet efficient S-box method based on chaotic sine map. Optik **130**, 1438–1444 (2017)
15. C. Wei-Bin, Z. Xin, Image encryption algorithm based on Henon chaotic system, in *2009 International Conference on Image Analysis and Signal Processing* (IEEE, Piscataway, 2009), pp. 94–97
16. L. Moysis, C. Volos, I. Stouboulos, S. Goudos, S. Çiç ek, V.T. Pham, V.K. Mishra, A novel chaotic system with application to secure communications, in *2020 9th International Conference on Modern Circuits and Systems Technologies (MOCAST)* (IEEE, Piscataway, 2020), pp. 1–4
17. L. Moysis, C. Volos, I. Stouboulos, S. Goudos, S. Çiçek, V.T. Pham, V.K. Mishra, A novel chaotic system with a line equilibrium: analysis and its applications to secure communication and random bit generation, in *Telecom* vol. 1, no. 3 (Multidisciplinary Digital Publishing Institute, Basel, 2020), pp. 283–296
18. A.V. Tutueva, E.G. Nepomuceno, A.I. Karimov, V.S. Andreev, D.N. Butusov, Adaptive chaotic maps and their application to pseudo-random numbers generation. Chaos Solitons Fract. **133**, 109615 (2020)
19. A. Tutueva, D. Butusov, Avoiding dynamical degradation in computer simulation of chaotic systems using semi-explicit integration: Rössler oscillator case. Fractal Fractional **5**(4), 214 (2021)

20. L. Moysis, A. Tutueva, C. Volos, D. Butusov, J.M. Munoz-Pacheco, H. Nistazakis, A two-parameter modified logistic map and its application to random bit generation. Symmetry **12**(5), 829 (2020)
21. A. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, A statistical test suite for random and pseudorandom number generators for cryptographic applications. Technical report, Booz-Allen and Hamilton Inc Mclean Va (2001)
22. A. Nag, J.P. Singh, S. Khan, S. Ghosh, S. Biswas, D. Sarkar, P.P. Sarkar, Image encryption using affine transform and XOR operation, in *2011 International Conference on Signal Processing, Communication, Computing and Networking Technologies* (IEEE, Piscataway, 2011), pp. 309–312
23. J. Ahmad, M.A. Khan, F. Ahmed, J.S. Khan, A novel image encryption scheme based on orthogonal matrix, skew tent map, and XOR operation. Neural Comput. Appl. **30**(12), 3847–3857 (2018)
24. L. Li, Y. Yao, X. Chang, Plaintext-dependent selective image encryption scheme based on chaotic maps and DNA coding, in 2017 International Conference on Dependable Systems and Their Applications (DSA) (IEEE, Piscataway, 2017), pp. 57–65
25. H.J. Stöckmann, *Quantum Chaos: An Introduction* (Cambridge University Press, Cambridge, 2000)
26. G. Casati, B. Chirikov, *Quantum chaos: between order and disorder*, (UK, Cambridge University Press, 2006)
27. K. Nakamura, T. Harayama, *Quantum Chaos And Quantum Dots*, vol. 3 (Oxford University Press, Oxford, 2004)
28. F. Musanna, S. Kumar, Image encryption using quantum 3-D Baker map and generalized gray code coupled with fractional Chen's chaotic system. Quant. Inf. Process. **19**(8), 1–31 (2020)
29. F. Faure, S. Nonnenmacher, On the maximal scarring for quantum cat map eigenstates. Commun. Math. Phys. **245**(1), 201–214 (2004)
30. M.E. Goggin, B. Sundaram, P.W. Milonn, Quantum logistic map. Phys. Rev. A **41**(10), 5705–5708 (1990). https://doi.org/10.1103/PhysRevA.41.5705
31. G. Ye, H. Wu, K. Jiao, D. Mei, Asymmetric image encryption scheme based on the Quantum logistic map and cyclic modulo diffusion. Math. Biosci. Eng. **18**(5), 5427–5448 (2021)
32. K. Jiao, G. Ye, Q. Mei, Image encryption scheme based on quantum logistic map and cellular automata, in *2021 IEEE 6th International Conference on Computer and Communication Systems (ICCCS)* (IEEE, Piscataway, 2021)
33. G. Ye, K. Jiao, X. Huang, B.M. Goi, W.S. Yap, An image encryption scheme based on public key cryptosystem and quantum logistic map. Sci. Rep. **10**(1), 1–19 (2020)
34. J. Xu, P. Li, F. Yang, H. Yan, High intensity image encryption scheme based on quantum logistic chaotic map and complex hyperchaotic system. IEEE Access **7**, 167904–167918 (2019)
35. A. Zaghloul, T. Zhang, M. Amin, A.A. Abd El-Latif, Color encryption scheme based on adapted quantum logistic map, in *Sixth International Conference on Digital Image Processing* (International Society for Optics and Photonics, Bellingham, 2014)
36. A. Akhshani, A. Akhavan, S.C. Lim, Z. Hassan, An image encryption scheme based on quantum logistic map. Commun. Nonlinear Sci. Numer. Simul. **17**(12), 4653–4661 (2012)
37. Z. Hua, B. Zhou, Y. Zhou, Sine chaotification model for enhancing chaos and its hardware implementation. IEEE Trans. Ind. Electron. **66**(2), 1273–1284 (2018)
38. X. Liu, D. Xiao, W. Huang, C. Liu, Quantum block image encryption based on Arnold transform and sine chaotification model. IEEE Access **7**, 57188–57199 (2019)
39. X. Wang, L. Liu, Y. Zhang, A novel chaotic block image encryption algorithm based on dynamic random growth technique. Opt. Lasers Eng. **66**, 10–18 (2015)
40. G. Ye, A block image encryption algorithm based on wave transmission and chaotic systems. Nonlinear Dynam. **75**(3), 417–427 (2014)
41. L. Xu, X. Gou, Z. Li, J. Li, A novel chaotic image encryption algorithm using block scrambling and dynamic index based diffusion. Opt. Lasers Eng. **91**, 41–52 (2017)

42. S.M. Seyedzadeh, B. Norouzi, M.R. Mosavi, S. Mirzakuchaki, A novel color image encryption algorithm based on spatial permutation and quantum chaotic map. Nonlinear Dynam. **81**(1), 511–529 (2015)
43. Y. Dong, X. Huang, Q. Mei, Y. Gan, Self-adaptive image encryption algorithm based on quantum logistic map. Secur. Commun. Netw. **2021**, 6674948 (2021)
44. G. Ye, H. Wu, K. Jiao, D. Mei, Asymmetric image encryption scheme based on the Quantum logistic map and cyclic modulo diffusion. Math. Biosci. Eng. **18**(5), 5427–5448 (2021)
45. J. He, S. Yu, J. Cai, Numerical analysis and improved algorithms for Lyapunov-exponent calculation of discrete-time chaotic systems. Int. J. Bifurcation Chaos **26**(13), 1650219 (2016)
46. R. Rhouma, S. Meherzi, S. Belghith, OCML-based colour image encryption. Chaos Solitons Fract. **40**(1), 309–318 (2009)
47. X. Wang, L. Liu, Y. Zhang, A novel chaotic block image encryption algorithm based on dynamic random growth technique. Opt. Lasers Eng. **66**, 10–18 (2015)
48. R.I. Abdelfatah, M. Nasr, M.A. Alsharqawy, Encryption for multimedia based on chaotic map: several scenarios. Multimedia Tools Appl. **79**(27), 19717–19738 (2020)
49. J.A. Núñez, P.M. Cincotta, F.C. Wachlin, Information entropy, in *Chaos in Gravitational N-Body Systems* (Springer, Dordrecht, 1996), pp. 43–53
50. J. Liang, Z. Shi, D. Li, M.J. Wierman, Information entropy, rough entropy and knowledge granulation in incomplete information systems. Int. J. Gen. Syst. **35**(6), 641–654 (2006)
51. Y. Wu, J.P. Noonan, S. Agaian, NPCR and UACI randomness tests for image encryption. Cyber J. Multidisciplinary J. Sci. Technol. J. Sel. Areas Telecommun. **1**(2), 31–38 (2011)
52. F. Özkaynak, Role of NPCR and UACI tests in security problems of chaos based image encryption algorithms and possible solution proposals, in *2017 International Conference on Computer Science and Engineering (UBMK)*. (IEEE, Piscataway, 2017), pp. 621–624
53. G. Alvarez, S. Li, Some basic cryptographic requirements for chaos-based cryptosystems. Int. J. Bifurcation Chaos **16**(8), 2129–2151 (2006)

Part III
Fractional Calculus

Chapter 6
On the Relationship Between Integer and Fractional PWL Systems with Multistable Behavior

H. E. Gilardi-Velázquez, J. L. Echenausia-Monroy, R. J. Escalante-González, B. B. Cassal-Quiroga, and G. Huerta-Cuellar

Abstract In this paper we present a study of the mechanisms that produce multistable behavior in integer and fractional Piece-Wise Linear (PWL) systems. The oscillator behavior is characterized using the Nearest Integer or $Round(x)$ function to control the switching processes and find the corresponding equilibria among the individual commutation surfaces. The integer system's path to multistability is controlled by a bifurcation parameter in the linear operator to modify the local stability, since for the fractional system the integration order is used as the bifurcation parameter, which gives rise to the same multistable phenomenon as in the integer order case. Both dynamical systems are studied using numerical simulations, bifurcation analysis, and Poincaré planes.

Keywords Multistability · Fractional systems · Multiscroll

H. E. Gilardi-Velázquez (✉) · B. B. Cassal-Quiroga
Facultad de Ingeniería, Universidad Panamericana, Aguascalientes, Aguascalientes, México
e-mail: hgilardi@up.edu.mx; bahiacassal@gmail.com

J. L. Echenausia-Monroy
Applied Physics Division, Center for Scientific Research and Higher Education at Ensenada, CICESE, Ensenada, B. C., Mexico
e-mail: echenausia@cicese.mx

R. J. Escalante-González
Electrical, Electronics and Mechatronics Department, Technological Institute of San Luis Potosí, San Luis Potosí, Mexico
e-mail: rodolfo.eg@slp.tecnm.mx

G. Huerta-Cuellar
Dynamical Systems Laboratory, CULagos, Universidad de Guadalajara, Centro Universitario de los Lagos, Lagos de Moreno, Jalisco, Mexico
e-mail: guillermo.huerta@academicos.udg.mx

G. Huerta Cuellar et al. (eds.), *Complex Systems and Their Applications*,
https://doi.org/10.1007/978-3-031-02472-6_6

6.1 Introduction

The phenomenon of multistability has attracted considerable interest in a variety of fields. This phenomenon has been found in integer-order multiscroll systems and has been studied in several papers using different mechanisms, i.e., by eigenspectra variation in a Piece-Wise Linear (PWL) system [1], by applying control techniques that produce the coexistence of multiple attractors [2, 3], or by changing the location of the equilibrium point by a bifurcation parameter that induces bistable behavior [4]. In fractional-order systems, the phenomenon of multistability has also been described [5–8], with these studies ranging from systems with no equilibrium points, to systems based on memristors, to systems that can have a family of bistable attractors.

Since the development of fractional calculus theory, the scientific community has been devoted to understanding the physical and geometrical relationships between fractional and integer systems. In recent decades, the study of chaotic systems using fractional-order derivatives has been intensively pursued [9–11], due to the characteristic properties of this type of systems, such as memory and inheritance, which represent several natural phenomena [12, 13]. Just as with integer order systems, there are many works in the literature dealing with the use of fractional order in chaotic systems, such as: The Duffing oscillator [14], the Chua system [15], the Rössler system [16], the Chen generator [17], the Lü model [18], among others, where it is proved that the non-integer systems exhibit the chaotic behavior [19–21]. However, there are few comparative analyzes in the literature that quantitatively show the relationships that exist between the behavior induced by the dynamics of a system when a fractional order is considered and the dynamics of the model when it is analyzed with an integration order equal to one [22].

In addition to recent studies, the possibilities of finding new behaviors and better descriptions of natural phenomena in fractional systems are a recurring theme in the literature [23–30]. In this sense, it was recently published that the use of fractional-order derivatives in PWL systems induces the occurrence of coexisting states that partition the basin of attraction of a monostable system into n possible basins, where n is the number of scrolls associated with the integer-order system [24], when the linear system is close to becoming stable by using fractional-order derivatives. In this work, the phenomenon presented is similar to that in integer systems [1], where the multistable behavior is achieved by a bifurcation parameter, this variation reaches critical eigenvalues for which the system becomes nearly stable, and a region where several single-scroll attractors coexist is generated. The occurrence of this phenomenon is correlated with the direction and location of the stable and unstable manifolds.

Motivated by the points described above, this paper presents a comparison of the effects that arise in the dynamics of a jerk system when the derivative order is used as a parameter versus a bifurcation parameter in an integer order system. The inherent properties of the systems, both integer and fractional, are analyzed using bifurcation diagrams such as the local maximum and the number of domains visited, Poincaré

planes in the switching surfaces. The results show a great similarity between the generated dynamics. This type of results enriches the analysis and understanding of the impact of using fractional derivatives.

6.2 Theory

6.2.1 Fractional Calculus

Fractional-order derivatives and integrals are generalizations of integer-order ones. Several different definitions for fractional-order derivatives can be found in the literature, the Riemann–Liouville and Caputo operators being the most important [31, 32]. The Caputo operator for fractional derivatives is defined by

$$D_0^q f(x) = \frac{1}{\Gamma(n-q)} \int_a^x \frac{f^{(n)}(t)}{(x-t)^{q-n+1}} dt, \tag{6.1}$$

with $n = \lceil q \rceil$, and Γ is the Gamma function defined as

$$\Gamma(z) = \int_0^\infty t^{z-1} e^{-t} dt. \tag{6.2}$$

A general commensurate fractional-order time-invariant system is described as follows:

$$D_0^{n_k} x(t) = f(t, x(t), D_0^{n_1} x(t), D_0^{n_2} x(t), \ldots, D_0^{n_{k-1}} x(t)), \tag{6.3}$$

subject to initial conditions

$$x^{(j)}(0) = x_0^{(j)}, \text{ with } j = 0, 1, \ldots, \lceil n_k \rceil - 1,$$

where $n_1, n_2, \ldots, n_k$ are rational numbers, such that $n_k > n_{k-1} > \cdots > n_1 > 0$, $n_j - n_{j-1} \leq 1$ for all $j = 2, 3, \ldots, k$ and $0 < n_1 \leq 1$. Let M be the least common multiple of the denominator of $n_1, n_2, \ldots, n_k$ and set $q = 1/M$ and $N = Mn_k$. Then Eq. (6.3) can be expressed in the following system of equations[31]:

$$\begin{aligned}
D_0^q x_0(t) &= x_1(t), \\
D_0^q x_1(t) &= x_2(t), \\
&\vdots \\
D_0^q x_{N-2}(t) &= x_{N-1}(t), \\
D_0^q x_{N-1}(t) &= f(t, x_0(t), x_{n1/q}(t), \ldots, x_{n_{k-1}/q}(t)),
\end{aligned}$$

with initial conditions

$$x_j(0) = \begin{cases} x_0^{(j/M)}, & \text{if } j/M \in \mathbb{N} \cup \{0\}, \\ 0, & \text{otherwise.} \end{cases}$$

Furthermore, this linear time-invariant system can be expressed in a matrix form as follows:

$$\frac{d^q \mathbf{x}(t)}{dt^q} = A\mathbf{x}, \tag{6.4}$$

where $\mathbf{x} \in \mathbb{R}^n$ represent the state vector, $A \in \mathbb{R}^{n \times n}$ a linear operator, and q is the fractional derivative order $0 < q < 1$.

One of the main features of fractional systems, as opposed to integer order systems, is that the stability of an equilibrium point depends on the derivative order (q) and creates a stability region, as shown in Fig. 6.1 [33]. As a result, the stability of an equilibrium point can be modified by the fractional derivative order. Considering a general N-dimensional fractional-order system, as described

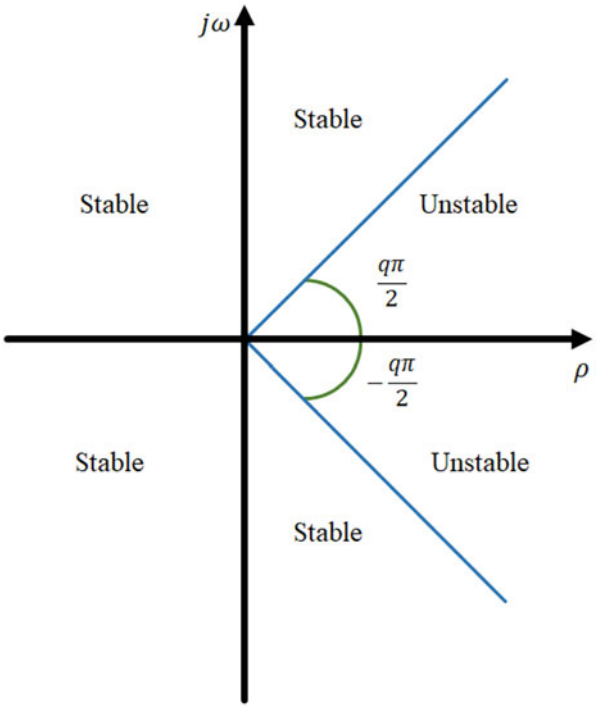


Fig. 6.1 Stability region of a fractional-order linear time-invariant system with order $0 < q < 1$

in Eq. (6.4), with λ_j eigenvalues, the equilibrium point stability of the system is defined by its eigenvalue analysis and described as follows [31]:

- The system is stable, if and only if $|\arg\lambda_j| \geq \frac{q\pi}{2}$.
- The system is asymptotically stable, if and only if $|\arg\lambda_j| > \frac{q\pi}{2}$.

The interest in this work is to obtain unstable equilibrium points, which implies that the equilibrium point given in Eq. (6.4) must have at least one eigenvalue located in the unstable region, *style* that the system does not fully satisfy the stability condition.

$$min|arg(\lambda_i)| > \frac{q\pi}{2}, \text{ for } i = 1, 2, \ldots, j. \tag{6.5}$$

6.2.2 Multiscroll Chaotic System

The study and evolution of chaotic systems capable of producing behaviors with greater dynamical richness by generating attractors with multiple scrolls has attracted much interest in recent decades. In the specific case, the jerk equation [34] is commonly used in multiscroll generators, since it can be expressed in the canonical controllable form [23]. There are several methods for generating chaotic attractors with multiple scrolls and the same number of equilibrium points [35–38]. This work focuses on chaotic attractors based on UDS theory and on the use of Piece-Wise Linear (PWL) functions [39–41].

In particular, it is considered the following family of affine linear systems:

$$\dot{X} = AX + B(X), \tag{6.6}$$

where $X = (x_1, x_2, x_3)^\top \in \mathbb{R}^3$ is the state vector, the real matrix $A \in \mathbb{R}^{3\times3}$ is nonsingular; and $B : \mathbb{R}^3 \to \mathbb{R}^3$ is a constant vector B_i in each domain D_i is determined by a switching function. We include the step function in the definition of these constant vectors B_i, $i = 1, \ldots, m$. Here, the matrix A is based on the linear ordinary differential equation (ODE) given by the jerk form: $\dddot{x} + a_{33}\ddot{x} + \dot{x}a_{32} + a_{31}x + \beta = 0$.

Definition 6.1 ([3]) Let $\Lambda = \{\lambda_1, \lambda_2, \lambda_3\}$ be the eigenspectrum of a linear operator $A\, in \mathbb{R}^{3\ times 3}$, such that $\sum_{i=1}^{3} \lambda_i < 0$, where λ_1 is a real number and λ_2, λ_3 are two complex conjugate numbers. A system given by the linear part of the system (6.6) is called UDS *Type I* if $\lambda_1 < 0$ and $Re\{\lambda_{2,3}\} > 0$; and it is *Type II* if $\lambda_1 > 0$ and $Re\{\lambda_{2,3}\} < 0$.

The position of the attractor depends on the following: the coefficient matrix A from the jerk equation [42] and the affine vector B as described by Gilardi-Velázquez et al. [1]:

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_1 & -a_2 & -a_3 \end{pmatrix}, \quad B(x) = \begin{pmatrix} 0 \\ 0 \\ \sigma(X) \end{pmatrix}; \tag{6.7}$$

where $a_1, a_2, a_3 \in \mathbb{R}$ and $\sigma(X) : \mathbb{R}^3 \to \mathbb{R}$ is the following step function, which generates a switching law to control the equilibria of the system:

$$\sigma(x) =$$

$$\begin{cases} b_1, & \text{if } x \in \mathcal{D}_1 = \{X \in \mathbb{R}^3 : \mathbf{v}^\top X < S_1\}; \\ b_2, & \text{if } x \in \mathcal{D}_2 = \{X \in \mathbb{R}^3 : S_1 \leq \mathbf{v}^\top X < S_2\}; \\ \vdots & \quad \vdots \\ b_m, & \text{if } x \in \mathcal{D}_m = \{X \in \mathbb{R}^3 : S_{m-1} \leq \mathbf{v}^\top X\}; \end{cases} \tag{6.8}$$

with $b_i \in \mathbb{R}$ (for $i = 1, \ldots, m$;) are the switching domains, $\mathbf{v} \in \mathbb{R}^3$ ($\mathbf{v} \neq 0$) a constant vector, and $\delta_1 \leq \delta_2 \leq \cdots \leq \delta_{m-1}$ determine the switching surfaces location.

Consider a multiscroll generator system based on the jerk equation [1, 43–45], which satisfies all conditions to be considered as UDS I for parameters a_i [1] as follows:

$$\begin{aligned} D^q x &= y, \\ D^q y &= z, \\ D^q z &= \alpha[-a_1 x - a_2 y - a_3 z - f(x)], \end{aligned} \tag{6.9}$$

where D^q is the Caputo operator of fractional order. The parameters a_i are constant values defining the eigenvalues of the system, where $a_i \in \mathbb{R}^+$. It is possible to generate an attractor with multiple scrolls by using a commutation law $f(x)$, in our case the Nearest Integer Function $Round(x)$, as shown in [1, 46], whose purpose is to partition the state space into D_i subdomains of equal size and add i equilibrium points to the system, which is achieved by coexisting many one-spiral unstable trajectories.

Since the systems described in Eq. (6.9) can be represented in terms of Eq. (6.6), the local stability for each equilibrium point is defined by the eigenvalues of the linear operator A, whose characteristic polynomial is $\lambda^3 + a_3\lambda^2 + a_2\lambda + a_1$. According to Proposition III.3. in [47], for $0 < a_1$, $0 < a_2 < a_1/a_2$, $0 < a_3$ the system is UDS type I for all the equilibrium points.

On the other hand, the local stability region of the fractional-order system depends mainly on two things, namely, the spectra of the system eigenvalues

related to the operating UDS region, i.e., the a_i parameters of Eq. (6.9), and on the integration order used to change the stability of the equilibrium point, since the multiscroll behavior is analyzed under a fractional order of $0.93 \leq q \leq 1$, which includes the critical integration order (q_c) with which it is possible to analyze the multiscroll generator for fixed parameters defined later without becoming stable when all state variables are considered with an equal integration order.

By considering a system based on the jerk equation (6.9), being $f(x)$ the Round function defined as

$$f(x) = C1 * Round(x/C2), \tag{6.10}$$

and for the following parameter values: $\alpha = 1$, $a_1 = 10.5$; $a_2 = 7$; $a_3 = 0.7$; $C1 = 0.9$; and $C2 = 0.6$. It is worth noting that these parameter values are the same ones used in [1]. With which it is possible to generate an attractor with multiple scrolls using Eq. (6.10) as commutation law, this induces an infinity of commutation surfaces and thus guarantees the generation of the same number of scrolls. In Fig. 6.2, the trajectory of the system Eq. (6.9) for the above parameters is shown in phase space (a), on the $x - y$, $x - z$ and $y - z$ planes (b), (c), and (d).

Consider D_i as each domain defined by the function *Round* and a Poincaré plane implemented exactly on the commutation surface around the origin. The Poincaré sections S_1, S_2 are implemented at $x = 0.3$ and $x = -0.3$, respectively, corresponding to the defined commutation values. A trajectory in D_i, with

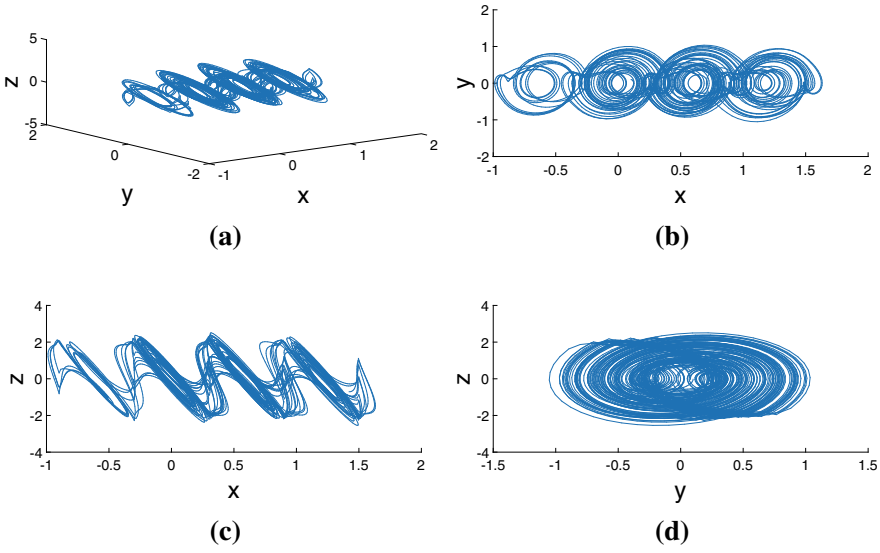


Fig. 6.2 Trajectory of the system (6.9) given by Eq. (6.10) (**a**) onto the phase space, in (**b**, **c**, **d**) projections onto the (x, y); (x, z); (y, z) respectively for $a_1 = 10.5$; $a_2 = 7$; $a_3 = 0.7$; $C1 = 0.9$; $C2 = 0.6$ $q = 1$ and $\alpha = 1$

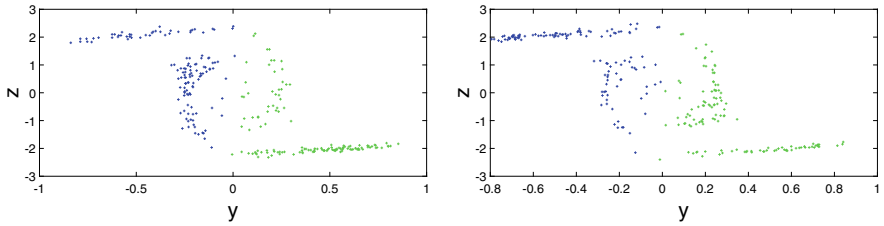


Fig. 6.3 Intersections of the trajectory of the system (6.9) with (6.10) with the commutation surface S_1 and S_2 (**b**) for $a_1 = 10.5$; $a_2 = 7$; $a_3 = 0.7$; $C1 = 0.9$; $C2 = 0.6$, $\alpha = 1$ and $q = 1$. The green marked points represent the crossing points from left to right and the blue marked points represent the crossing points from right to left for each plane

equilibrium point $X_i^* \in \mathcal{D}_i$. As the trajectory oscillates around X_i^*, the distance to the equilibrium point increases leaving the current domain $\mathcal{D}_i$ to $\mathcal{D}_{i+1}$ Fig. 6.3a (or $\mathcal{D}_i$ to $\mathcal{D}_{i-1}$ Fig. 6.3b) by the commutation surface near the intersection of the unstable manifold with the Poincaré plane, as can be seen from the intersections of the trajectory ϕ^{t_j} in Fig. 6.3. The green crossed points represent the intersection from left to right and the blue crossed points represent the intersection from right to left for each plane considered. Note that most trajectories are equally distributed in both directions.

6.3 Multistability in the PWL System Through Bifurcation Parameter

In this work, we consider an oscillator that has already been reported to exhibit multistability behavior: [1]. Considering the system defined in Eq. (6.9) for integer order and the parameters a_i defined before, we now use the parameter α as a bifurcation parameter that allows to change the local stability and the directions of the stable and unstable manifolds. Figure 6.4a shows the bifurcation diagram of the local maximum along x for each scroll at $\mathcal{D}_i$ for the range of parameter $0 \leq \alpha \leq 1.5$. In addition, Fig. 6.4b shows the number of domains $\mathcal{D}_i$ visited by the system for equal values of α. Note that the number of domains $\mathcal{D}$ visited remains at the constant value of 3 when the range of $\alpha \geq 1.25$, since the system has only one attractor located in the inside domain, as shown in Fig. 6.5 [1]. Both diagrams were calculated with the same initial condition $\mathbf{X}_0 = (0.1, 0.1, 0)^T$.

Considering the values used in the previous section and $\alpha = 1.42$, the system leads to a multistable state phenomenon due to the round function and the location of its eigenvectors [1], as shown in Fig. 6.4 for the initial condition $\mathbf{X}_0 = (0.1, 0.1, 0)^T$. The attractor is located near the equilibria at the origin due to the given initial condition. However, there is no oscillation near the neighboring equilibrium points.

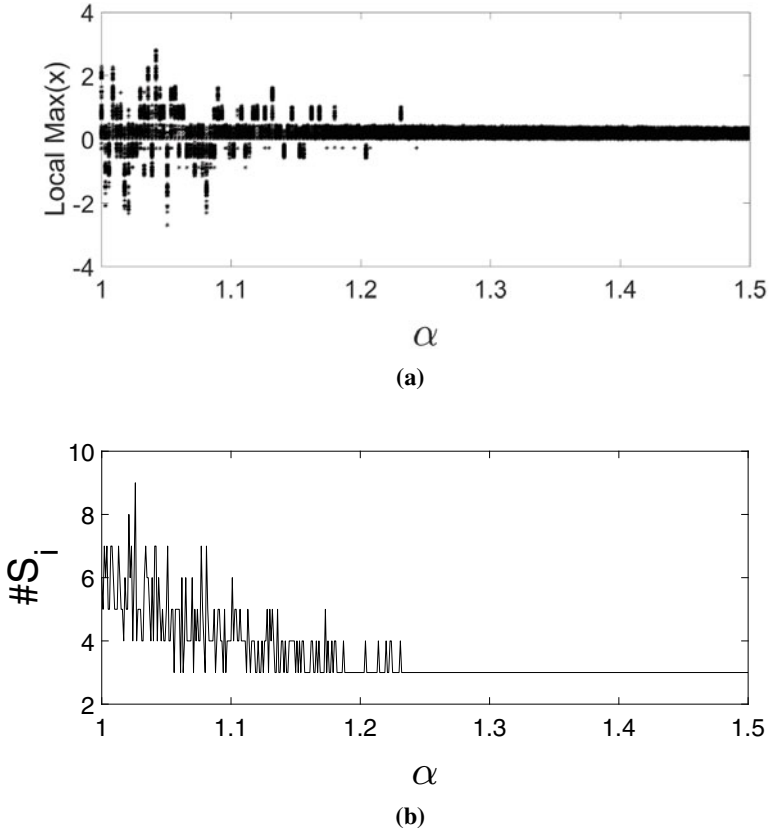


Fig. 6.4 (**a**) Bifurcation diagram of the system given by the Eqs. (6.9) with (6.10) for $a_1 = 10.5$; $a_2 = 7$; $a_3 = 0.7$; $C1 = 0.9$; $C2 = 0.6$, $q = 1.0$, for the value of $1 \leq \alpha \leq 1.5$. Figure (**b**) shows the number of domains $\mathcal{D}_i$ visited by the trajectory of the system for the same values of the above bifurcation parameters. The initial condition considered for both diagrams is $\mathbf{X}_0 = (0.1, 0.1, 0)$

The reason for this lies in the eigenvectors, as they are not located in the same way as in the previous section for $\alpha = 1$, i.e., the stable manifold of the domain in which the trajectory oscillates does not coincide with the stable manifolds of the neighboring domains [1].

This can be easily observed in Fig. 6.5, which is now described. Figure 6.5b and d show the projection of a trajectory onto the plane (x, y), both initialized with the same initial condition near the origin $\mathbf{X}_0 = (0.1, 0.1, 0)^T$. Note in the projections that as time increases, the oscillating clockwise system trajectory becomes larger until eventually the trajectories on the scrolls cross the commutation surface. Obviously, the trajectories of both systems cross the commutation surfaces S_1, S_2 near the intersection of the unstable and stable manifolds. Considering the

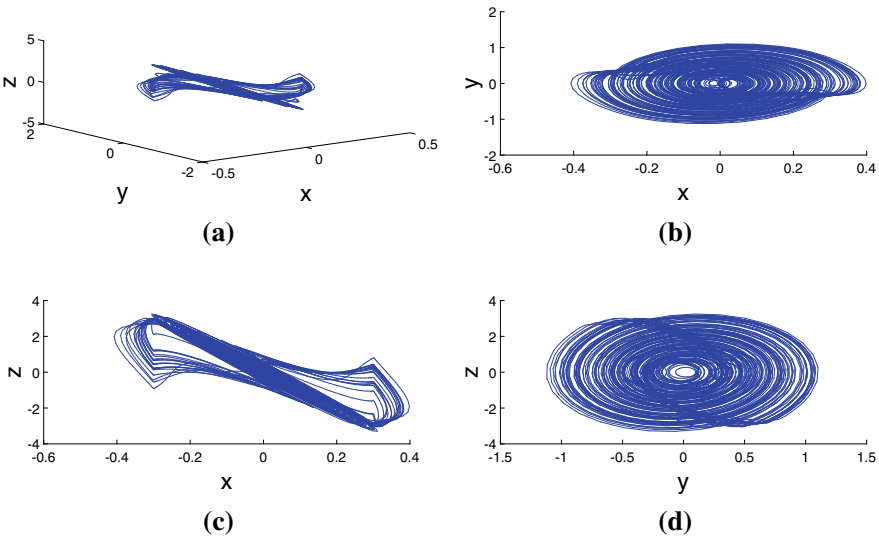


Fig. 6.5 Trajectory of the system (6.9) given by Eq. (6.10) (**a**) on the phase space, (**b**, **c**, **d**) projections on the (x, y); (x, z); (y, z)-plane, for $a_1 = 10.5$; $a_2 = 7$; $a_3 = 0.7$; $C1 = 0.9$; $C2 = 0.6$ $q = 1$ and $\alpha = 1.42$

different projections of the attractors, for example the projection onto the planes (x, z) in Fig. 6.5c and d, it can be seen that the trajectory returns in the direction of the equilibrium point associated with the domain $x = \pm 0.3$ in Fig. 6.5b, c and $y = \pm 0.5$, $z = \pm 3.0$. Which was slightly changed by varying the parameter from $\alpha = 1$ to $\alpha = 1.42$.

Now, to understand the behavior of the trajectory, a Poincaré plane was implemented exactly on the commutation surface, as described in the previous section. The crossing events of interest are the identification of the crossing points, or in other words, the escape points and the return points to the origin domain $\mathcal{D}_i$ to $\mathcal{D}_{i+1}$ or $\mathcal{D}_i$ to $\mathcal{D}_{i-1}$. Figure 6.6 shows the intersection points of the trajectories for the system Eq. (6.9) with $\alpha = 1.42$ for the right and left planes, respectively, in red asterisk the intersections from right to left and in black asterisk the intersection from left to right, in green and blue crosses the intersection points for $\alpha = 1$ described in the previous section.

Note that for the right plane Fig. 6.5a, the exit points of this domain are shown in red and many crossing points have been lost compared to the multiscroll attractor ($\alpha = 1$), just as many crossing points have been lost for the return points shown in black. The same characteristic can be observed in a symmetrical way for the left plane Fig. 6.5b, where the exit points are shown in black color and the return points in red. Since this work is not concerned with explaining the emergence of multistable dynamics, a more detailed explanation of this phenomenon can be found in reference [1].

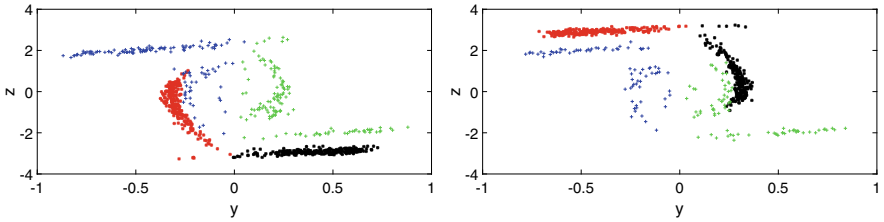


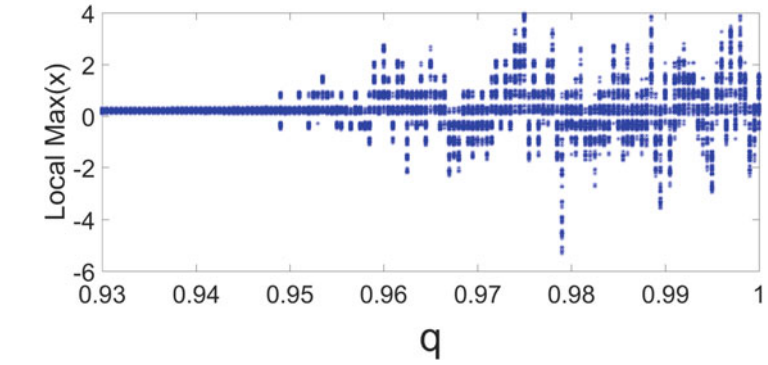
Fig. 6.6 Intersections of the trajectory of the system (6.9) with (6.10) with the commutation surface S_1 and S_2, for $a_1 = 10.5$; $a_2 = 7$; $a_3 = 0.7$; $C1 = 0.9$; $C2 = 0.6$, and $q = 1$ (**a**), (**b**), respectively. In green cross are marked the points representing the intersections from left to right and in blue cross those from right to left for each plane for $\alpha = 1$. In red asterisk the crossings events from right to left and in black asterisk those from left to right for $\alpha = 1.42$

6.4 Multistability in the PWL System Through Derivative Order

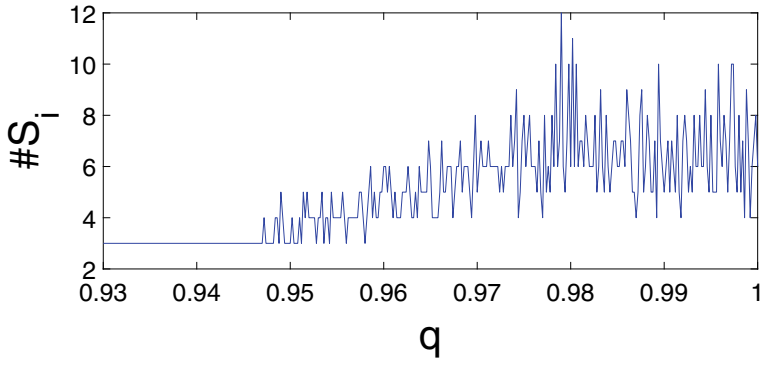
According to the reference [24], it is possible to generate multistable behavior in PWL systems by fractional derivatives. Following the mechanism described in [24], the system described by Eq. (6.9) is now analyzed numerically by using the Adams–Bashforth–Moulton (ABM) method [48] for changes in integration order q. The algorithm was constructed as a generalization of the classical ABM integrator, which is well known in solving problems with first-order switching systems [33, 49]. Considering the stability constraints for fractional-order systems, the algorithm designed for Eq. (6.9) a critical integration order at $q_c = 0.927$ to preserve local instability. According to the stability theory for fractional systems [33], if the system is analyzed with orders of integration $q < q_c$, the dynamics of the system will be stable, turning each equilibrium point into a focus attractor.

The system of fractional order, which is described in Eq. (6.9), is analyzed for a fixed bifurcation parameter $\alpha = 1.0$, considering the switching function described by $Round(x)$, an integration step size of $h = 0.01$, and considering the same integration order in the three state variables. As in the previous examples, a bifurcation diagram of fractional order vs. local maximum in the state variable x is shown in Fig. 6.7. The qualitative changes in the system dynamics when the derivative order is reduced are depicted in Fig. 6.7. This behavior is similar to that computed for the integer order system. In addition, Fig. 6.7b depicts the number of domains $\mathcal{D}_i$ visited by the system for equal values of the derivative order q. Note that the number of domains $\mathcal{D}$ visited remains at the constant value of 3 when the range of $q \leq 0.945$ approximately because the system has only one scroll in the domain around the origin, as shown in Fig. 6.8.

As in the case of the integer order system with changes in α, in Fig. 6.8b and (d) a projection of a trajectory onto the plane (x, y) and (y, z), both initialized with the same initial condition near the origin $\mathbf{X}_0 = (0.1, 0.1, 0)^T$, is shown. Note that



(a)



(b)

Fig. 6.7 (**a**) Bifurcation diagram of the system given by Eqs. (6.9) with (6.10) for $a_1 = 10.5$; $a_2 = 7$; $a_3 = 0.7$; $C1 = 0.9$; $C2 = 0.6$, $\alpha = 1.0$, for the value of $0.93 \leq q \leq 1.0$. Figure (**b**) shows the number of domains $\mathcal{D}_i$ visited by the trajectory of the system for the same values of the bifurcation parameters above. The initial condition considered for both diagrams is $\mathbf{X}_0 = (0.1, 0.1, 0)^T$

in the projections, the clockwise trajectory of the oscillating system increases with time until eventually the trajectories on the scroll cross the commutation surface. Obviously, the trajectories of both systems cross the commutation surface S_1 and S_2 near the intersection of the unstable and stable manifolds. Considering the different projections of the attractor, for example the projection onto the planes (x, z), in Fig. 6.8c and d it can be seen that the trajectory returns to the equilibrium point associated with the domain at $x = \pm 0.3$ in Fig. 6.8b, c and $y = \pm 0.5$, $z = \mp 3.0$. This was slightly modified by changing the parameter from $q = 1$ to $q = 0.945$.

Poincaré planes are used to graphically represent the intersection points between domains where the scroll is generated by the switching surfaces[1]. A Poincaré plane was implemented exactly at the commutation surface, as described in the

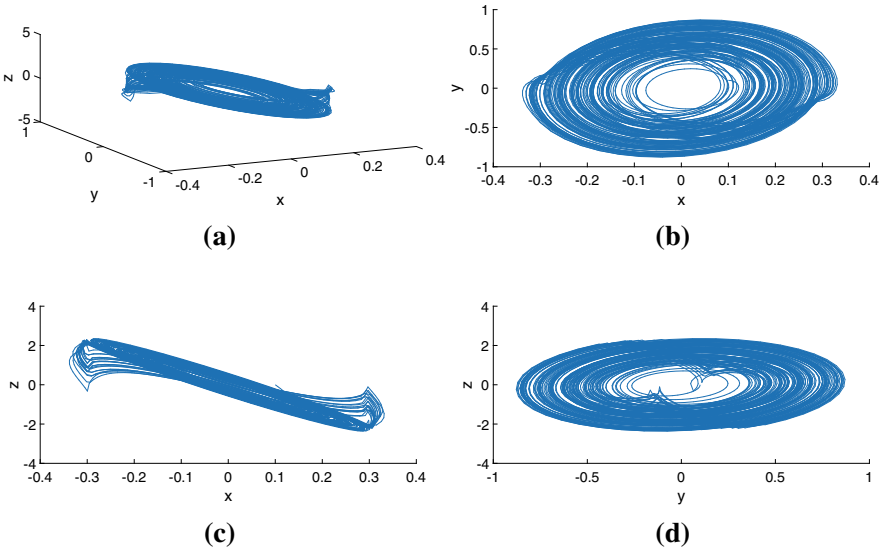


Fig. 6.8 Trajectory of the system (6.9) given by Eq. (6.10) (**a**) onto the phase space, in (**b**, **c**, **d**) projections onto the (x, y); (x, z); (y, z) respectively for $a_1 = 10.5$; $a_2 = 7$; $a_3 = 0.7$; $C1 = 0.9$; $C2 = 0.6$ $q = 0.945$, and $\alpha = 1$

previous section. The interest is the identification of the crossing point, or in other words, the escape points and return points to the origin domain $\mathcal{D}_i$ to $\mathcal{D}_{i+1}$ or $\mathcal{D}_i$ to $\mathcal{D}_{i-1}$. Figure 6.9 shows the trajectory intersection points for the system (6.9) with $q = 0.945$ for the right and left planes, respectively; red asterisks show the intersections from right to left and the black asterisks show the crossing points from left to right; green and blue crosses show the intersection points for $q = 1$ and $\alpha = 1$, respectively, described in Sect. 6.2.2.

As in the previous section, in the right plane Fig. 6.9a the exit points of this domain are depicted in red, and compared with the multiscroll attractor ($q = 1.0$), many crossing points were lost, as well as for the return points depicted in black many crossing points were lost. This same characteristic can be observed in a symmetric way for the left plane Fig. 6.9b, where the exit points are represented by the black color and the return points in red.

Since the goal of analyzing the multistable behavior with Poincaré planes is to identify the relationship between the integer and fractional systems and to analyze the behavior of each attractor at the switching surfaces, the crossing points in each plane for the multistable attractors were determined. The results are shown in Figs. 6.4, 6.7, 6.6, and 6.9. The bifurcation analysis shows a great similarity in the evolution of the dynamics under the parametric changes considered in each case. Although many crossing points were lost in both cases, it is important to note in the Poincaré plane analysis that for the fractional case the return points remain aligned

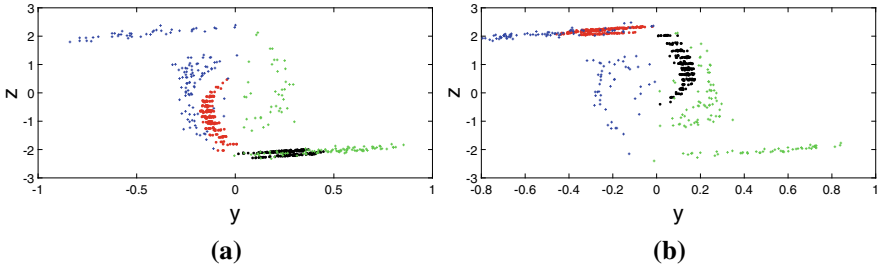


Fig. 6.9 Intersections of the trajectory of the system (6.9) with (6.10) with the commutation surface S_1 and S_2, for $a_1 = 10.5$; $a_2 = 7$; $a_3 = 0.7$; $C1 = 0.9$; $C2 = 0.6$, and $\alpha = 1$ (**a**), (**b**). In green cross the points representing the intersections from left to right and in blue cross those from right to left for each plane for $q = 1$. In red asterisk the intersections from right to left and in black asterisk the intersection from left to right for $q = 0.945$

with the return points of the multiscroll attractor, i.e., the multiscroll attractor and the fractional multistable attractor share common crossing points.

6.5 Conclusions

In this work it has been presented the study of generating multistable behavior in a multiscroll system using the derivative order as a parameter versus a bifurcation parameter in an integer system. Using the Nearest Integer or $round(x)$ function for the PWL systems considered in the UDS of Type I theory, in the case of the integer system the phenomenon of multistability was driven by the changes in the bifurcation parameter α, while for the fractional system the generation of multistable behavior was achieved by using the integration order as the system parameter. Both dynamics were analyzed by bifurcation diagrams and Poincaré planes, showing that the mechanism generating the multistable phenomenon is similar. However, it is shown that the dynamics obtained by the two mechanisms are not equivalent, since the main difference between the two systems lies in the Poincaré analysis. For the integer system, the return points are shifted and for the fractional case they remain aligned with the return points of the multiscroll attractor.

References

1. H. Gilardi-Velázquez, L. Ontañón-García, D. Hurtado-Rodriguez, E. Campos-Cantón, Multistability in piecewise linear systems versus eigenspectra variation and round function. Int. J. Bifurcation Chaos **27**(09), 1730031 (2017). https://doi.org/10.1142/S0218127417300312
2. L. Ontañón-García, E. Campos-Cantón, Widening of the basins of attraction of a multistable switching dynamical system with the location of symmetric equilibria. Nonlinear Analy. Hybrid Syst. **26**, 38–47 (2017)

3. H.E. Gilardi-Velázquez, R.d.J. Escalante-González, E. Campos-Cantón, Bistable behavior via switching dissipative systems with unstable dynamics and its electronic design, IFAC-PapersOnLine **51**(13), 502–507 (2018)
4. L. Chua, The genesis of chua's circuit. Arch. Elektron. Uebertrag. **46**(3), 250–257 (1992)
5. V. Pham, S. Kingni, C. Volos, S. Jafari, T. Kapitaniak, A simple three-dimensional fractional-order chaotic system without equilibrium: dynamics, circuitry implementation, chaos control and synchronization. AEU-Int. J. Electron. Commun. **78**, 220–227 (2017)
6. Y. Xu, K. Sun, S. He, L. Zhang, Dynamics of a fractional-order simplified unified system based on the adomian decomposition method. Eur. Phys. J. Plus **131**(6), 186 (2016)
7. J. Munoz-Pacheco, E. Zambrano-Serrano, C. Volos, S. Jafari, J. Kengne, K. Rajagopal, A new fractional-order chaotic system with different families of hidden and self-excited attractors. Entropy **20**(8), 564 (2018)
8. C. Zhou, Z. Li, F. Xie, Coexisting attractors, crisis route to chaos in a novel 4d fractional-order system and variable-order circuit implementation. Eur. Phys. J. Plus **134**(2), 73 (2019)
9. I. Petráš, *Fractional-Order Nonlinear Systems: Modeling, Analysis and Simulation* (Springer Science & Business Media, Berlin, 2011). https://doi.org/10.1007/978-3-642-18101-6
10. T. Freeborn, B. Maundy, A. Elwakil, Measurement of supercapacitor fractional-order model parameters from voltage-excited step response. IEEE J. Emerg. Sel. Top. Circuits Syst. 3(3), 367–376 (2013). https://doi.org/10.1109/JETCAS.2013.2271433
11. A. Elwakil, Fractional-order circuits and systems: An emerging interdisciplinary research area. IEEE Circuits Syst. Mag. 10(4), 40–50 (2010). https://doi.org/10.1109/MCAS.2010.938637
12. C. Li, F. Zhang, J. Kurths, F. Zeng, Equivalent system for a multiple-rational-order fractional differential system. Phil. Trans. R. Soc. A: Math. Phys. Eng. Sci. **371**(1990), 20120156 (2013). https://doi.org/10.1098/rsta.2012.0156
13. C. Li, Y. Chen, J. Kurths, Fractional calculus and its applications (2013). https://doi.org/10.1098/rsta.2013.0037
14. X. Gao, J. Yu, Chaos in the fractional order periodically forced complex duffing's oscillators. Chaos, Solitons & Fractals **24**(4), 1097–1104 (2005). https://doi.org/10.1016/j.chaos.2004.09.090
15. D. Cafagna, G. Grassi, Observer-based synchronization for a class of fractional chaotic systems via a scalar signal: results involving the exact solution of the error dynamics. Int. J. Bifurc. Chaos **21**(03), 955–962 (2011). https://doi.org/10.1142/S021812741102874X
16. C. Li, G. Chen, Chaos and hyperchaos in the fractional-order rössler equations. Phys. A Stat. Mech. Appl. **341**, 55–61 (2004). https://doi.org/10.1016/j.physa.2004.04.113.
17. C. Li, G. Peng, Chaos in chen's system with a fractional order. Chaos Solitons Fractals **22**(2), 443–450 (2004). https://doi.org/10.1016/j.chaos.2004.02.013
18. J. Lu, Chaotic dynamics of the fractional-order lü system and its synchronization. Phys. Lett. A **354**(4), 305–311 (2006) https://doi.org/10.1016/j.physleta.2006.01.068
19. I. Grigorenko, E. Grigorenko, Chaotic dynamics of the fractional lorenz system. Phys. Rev. Lett. **91**(3), 034101 (2003). https://doi.org/10.1103/PhysRevLett.91.034101
20. M.R. Faieghi, H. Delavari, Chaos in fractional-order genesio–tesi system and its synchronization. Commun. Nonlinear Sci. Num. Simul. **17**(2), 731–741 (2012). https://doi.org/10.1016/j.cnsns.2011.05.038
21. L.-G. Yuan, Q.-G. Yang, Parameter identification and synchronization of fractional-order chaotic systems. Commun. Nonlinear Sci. Num. Simul. **17**(1), 305–316 (2012). https://doi.org/10.1016/j.cnsns.2011.04.005
22. S. Dadras, H. Momeni, A novel three-dimensional autonomous chaotic system generating two, three and four-scroll attractors. Phys. Lett. A **373**(40), 3637–3642 (2009). https://doi.org/10.1016/j.physleta.2009.07.088
23. J.L. Echenausía-Monroy, H. Gilardi-Velázquez, R. Jaimes-Reátegui, V. Aboites, G. Huerta-Cuéllar, A physical interpretation of fractional-order-derivatives in a jerk system: electronic approach. Commun. Nonlinear Sci. Num. Simul. **90**, 105413 (2020). https://doi.org/10.1016/j.cnsns.2020.105413

24. J.L. Echenausía-Monroy, G. Huerta-Cuellar, R. Jaimes-Reátegui, J.H. García-López, V. Aboites, B.B. Cassal-Quiroga, H.E. Gilardi-Velázquez, Multistability emergence through fractional-order-derivatives in a pwl multi-scroll system. Electronics **9**(6), 880 (2020). https://doi.org/10.3390/electronics9060880
25. J.M. Munoz-Pacheco, C. Posadas-Castillo, E. Zambrano-Serrano, The effect of a non-local fractional operator in an asymmetrical Glucose-Insulin regulatory system: analysis, synchronization and electronic implementation. Symmetry **12**(9), 1395 (2020)
26. E.-Z. Dong, R.-H. Li, S.-Z. Du, A multi-directional controllable multi-scroll conservative chaos generator: modelling, analysis, and FPGA implementation. Chin. Phys. B **30**(2), 020505 (2021)
27. P.Y. Dousseh, C. Ainamon, C.H. Miwadinou, A.V. Monwanou, J.B. Chabi Orou, Chaos in a financial system with fractional order and its control via sliding mode. Complexity **2021**, 4636658 (2021)
28. J. Ramadoss, S. Aghababaei, F. Parastesh, K. Rajagopal, S. Jafari, I. Hussain, Chimera state in the network of fractional-order fitzhugh–nagumo neurons, Complexity **2021** (2021)
29. J.M. Munoz-Pacheco, L.C. Lujano-Hernández, C. Muñiz-Montero, A. Akgül, L.A. Sánchez-Gaspariano, C.-B. Li, M. Çağri Kutlu, Active realization of fractional-order integrators and their application in multiscroll chaotic systems, Complexity **2021**, 6623855 (2021)
30. Z.-A.S. Rahman, B.H. Jasim, Y.I. Al-Yasir, R.A. Abd-Alhameed, B.N. Alhasnawi, A new no equilibrium fractional order chaotic system, dynamical investigation, synchronization, and its digital implementation. Inventions **6**(3), 49 (2021)
31. K. Diethelm, *The Analysis of Fractional Differential Equations: An Application-Oriented Exposition Using Differential Operators of Caputo Type* (Springer Science & Business Media, Berlin, 2010). https://doi.org/10.1007/978-3-642-14574-2
32. I. Podlubny, *Fractional Differential Equations: An Introduction to Fractional Derivatives, Fractional Differential Equations, to Methods of Their Solution and Some of Their Applications*, vol. 198 (Elsevier, Amsterdam, 1998)
33. E. Zambrano-Serrano, E. Campos-Cantón, J. Muñoz-Pacheco, Strange attractors generated by a fractional order switching system and its topological horseshoe. Nonlinear Dyn. **83**(3), 1629–1641 (2016). https://doi.org/10.1007/s11071-015-2436-z
34. J. Sprott, Some simple chaotic flows. Phys. Rev. E **50**(2), R647 (1994). https://doi.org/10.1103/PhysRevE.50.R647
35. M. Yalcin, J. Suykens, J. Vandewalle, Experimental confirmation of 3-and 5-scroll attractors from a generalized chua's circuit. IEEE Trans. Circuits Syst. I Fund. Theory Appl. **47**(3), 425–429 (2000). https://doi.org/10.1109/81.841929
36. J. Suykens, A. Huang, A family of n-scroll attractors from a generalized chua's circuit. Archiv fur Elektronik und Ubertragungstechnik **51**(3), 131–137 (1997)
37. R. Newcomb, N. El-Leithy, Chaos generation using binary hysteresis. Circuits Syst. Sig. Proce. **5**(3), 321–341 (1986). https://doi.org/10.1007/BF01600066
38. J. Lü, F. Han, X. Yu, G. Chen, Generating 3-d multi-scroll chaotic attractors: A hysteresis series switching method. Automatica **40**(10), 1677–1687 (2004). https://doi.org/10.1016/j.automatica.2004.06.001
39. M. Yalcin, S. Ozoguz, J. Suykens, J. Vandewalle, n-scroll chaos generators: a simple circuit model. Electron. Lett. **37**(3), 147–148 (2001). https://doi.org/10.1049/el:20010114
40. W. Tang, G. Zhong, G. Chen, K. Man, Generation of n-scroll attractors via sine function. IEEE Trans. Circuits Syst. I Fund. Theory Appl. **48**(11), 1369–1372 (2001). https://doi.org/10.1109/81.964432
41. J. Suykens, J. Vandewalle, Generation of n-double scrolls (n= 1, 2, 3, 4,...), IEEE Trans. Circuits Syst. I Fund. Theory Appl. **40**(11), 861–867 (1993). https://doi.org/10.1109/81.251829
42. E. Campos-Cantón, J. Barajas-Ramirez, G. Solis-Perales, R. Femat, Multiscroll attractors by switching systems. Chaos Interdiscip. J. Nonlinear Sci. **20**(1), 013116 (2010)
43. J. Echenausía-Monroy, J. García-López, R. Jaimes-Reátegui, D. López-Mancilla, G. Huerta-Cuellar, Family of bistable attractors contained in an unstable dissipative switching system associated to a SNLF. Complexity **2018** (2018). https://doi.org/10.1155/2018/6794791

44. E. Campos-Cantón, Chaotic attractors based on unstable dissipative systems via third-order differential equation. Int. J. Modern Phys. C **27**(01), 1650008 (2016). https://doi.org/10.1142/S012918311650008X
45. J. Echenausía-Monroy, G. Huerta-Cuellar, A novel approach to generate attractors with a high number of scrolls. Nonlinear Analy. Hybrid Syst. **2019**, 100822 (2019). https://doi.org/10.1016/j.nahs.2019.100822
46. G. Huerta-Cuellar, E. Jimenez-Lopez, E. Campos-Cantón, A. Pisarchik, An approach to generate deterministic brownian motion. Commun. Nonlinear Sci. Num. Simul. **19**(8), 2740–2746 (2014)
47. A. Anzo-Hernández, H.E. Gilardi-Velázquez, E. Campos-Cantón, On multistability behavior of unstable dissipative systems. Chaos Interdiscip. J. Nonlinear Sci. **28**(3), 033613 (2018)
48. K. Diethelm, N. Ford, A. Freed, A predictor-corrector approach for the numerical solution of fractional differential equations. Nonlinear Dyn. **29**(1–4), 3–22 (2002). https://doi.org/10.1023/A:1016592219341
49. H. Gilardi-Velázquez, E. Campos-Cantón, Nonclassical point of view of the brownian motion generation via fractional deterministic model. Int. J. Modern Phys. C **29**(03), 1850020 (2018). https://doi.org/10.1142/S0129183118500201.

Chapter 7
Approximation of Fractional-Order Controllers for Mechatronic Applications

Stavroula Kapoulea, Costas Psychalinos, and Ahmed S. Elwakil

Abstract Two different approximation tools for deriving integer-order rational transfer functions, which fulfill the loop-shaping requirements in fractional-order (FO) controllers used in mechatronic applications, are presented and compared in this work. The first one is based on the employment of a curve fitting-based tool offered by the MATLAB symbolic toolbox, while the second one is based on the utilization of the FLOreS toolbox, also available in MATLAB. Both methods approximate the magnitude and phase frequency responses derived from the original fractional-order transfer function. These tools are compared with regards to the offered accuracy and, also, from the implementation point of view. This is performed through a design example, where a planar precision positioning mechatronic system is used as plant and the behavior of the system is evaluated through simulation results, using OrCAD PSpice simulator.

Keywords Fractional-order controllers · Loop shaping · Curve-fitting approximation · FLOreS toolbox · Mechatronic systems

This research is co-financed by Greece and the European Union (European Social Fund-ESF) through the Operational Programme "Human Resources Development, Education and Lifelong Learning" in the context of the project "Strengthening Human Resources Research Potential via Doctorate Research-2nd Cycle" (MIS-5000432), implemented by the State Scholarships Foundation (IKY).

S. Kapoulea (✉) · C. Psychalinos
University of Patras, Physics Department, Electronics Laboratory, Rio Patras, Greece
e-mail: skapoulea@upnet.gr; cpsychal@upatras.gr

A. S. Elwakil
Department of Electrical and Computer Engineering, College of Engineering, University of Sharjah, Sharjah, United Arab Emirates - AE
e-mail: elwakil@ieee.org

G. Huerta Cuellar et al. (eds.), *Complex Systems and Their Applications*,
https://doi.org/10.1007/978-3-031-02472-6_7

7.1 Introduction

Loop shaping is a method for designing controllers in the frequency domain, through the adjustment of the open-loop frequency response of the system controller plant, in order to compromise the conflict between the following requirements: (a) the high open-loop gain at low frequencies to ensure good tracking and disturbance rejection and low gain at high frequencies to ensure noise suppression and (b) the phase margin at crossover frequency to guarantee stability of the system. In other words, a trade-off between precision and bandwidth versus stability and robustness must be performed through the design framework of the controller [1–5]. Fractional-order controllers are attractive schemes for performing loop shaping, and this is originated from the fact that the order of both the integration and differentiation stages non-integer, in contrast to their inter-order counterparts where the order is fixed and equal to one. This opens the door for independent adjustment of the slopes of the open-loop gain and frequency responses allowing the fine-tuning of the desired characteristics of the system [6–12]. This is performed at the expense of the increased circuit complexity, because the crucial parts of the implementation of fractional-order differentiators and integrators are fractional-order capacitors which are not available in the market. This requires the development of networks that approximate their behavior, and a quick solution is the utilization of the Cauer and Foster networks for substituting the fractional-order capacitors [13]. Another solution is based on the approximation of the transfer function of the fractional-order integrator/differentiator using approximation tools for deriving rational integer-order transfer functions [14].

The problem of both aforementioned approaches appears in the cases where the transfer functions which must be synthesized have not the classical form of the powers of the Laplacian operator. Two possible solutions for overcoming this obstacle are the employment of the *frd* and *fitfrd* functions [15] or of the FLOreS toolbox [16], both available in the MATLAB software.

The goal of this work is the systematic comparison of the aforementioned tools with regard to the approximation accuracy level, as well as to the complexity of the required implementations. The chapter is organized as follows: the two approximation tools are presented in Sect. 7.2, considering the case of a control system suitable for a positioning mechatronic system. The implementation issues are discussed in Sect. 7.3, while in Sect. 7.4 simulation results, derived using the AD844 discrete component ICs, are presented.

7.2 Approximation Tools

As a design example, let us consider the control loop of Fig. 7.1, where a three-degree-of-freedom planar precision positioning mechatronic system is used as plant.

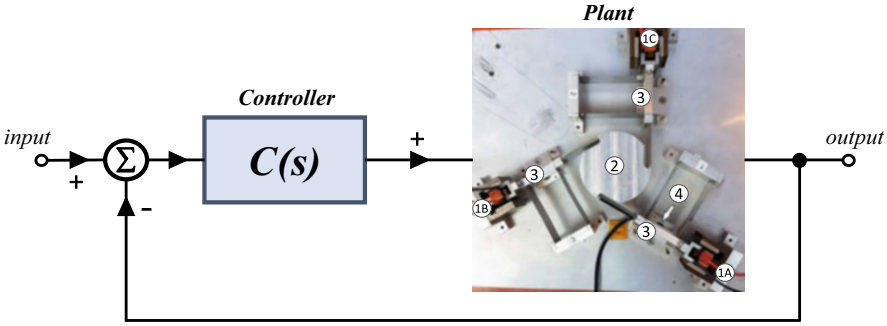


Fig. 7.1 Control loop of the three-degree-of-freedom planar precision positioning system

The transfer function that describes the plant behavior is given by [17]

$$P(s) = \frac{1.429 \times 10^8}{175.9s^2 + 7738s + 1.361 \times 10^6}. \tag{7.1}$$

The general form of a three-stage controller is given by [17]

$$C(s) = K_P \cdot \left(1 + \frac{\omega_i}{s}\right)^{\lambda} \cdot \left(\frac{1 + \frac{s}{\omega_d}}{1 + \frac{s}{\omega_t}}\right)^{\mu} \cdot \left(\frac{1}{1 + \frac{s}{\omega_l}}\right)^{\gamma}. \tag{7.2}$$

The term $K_P \cdot \left(1 + \frac{\omega_i}{s}\right)^{\lambda}$ is the PI/phase lag part, with ω_i being the unity gain frequency of the integration part, while the term $\left(1+\frac{s}{\omega_d}/1+\frac{s}{\omega_t}\right)^{\mu}$ represents the phase lead part, with ω_d and ω_i being the zero and pole frequency, respectively. The remaining term $\left(1/1+\frac{s}{\omega_l}\right)^{\gamma}$ is a lowpass (LP) filter with ω_l being its cutoff frequency.

The transfer function in Eq. (7.2) cannot be directly realized by conventional elements, and therefore, its efficient approximation is a crucial part of the design procedure. The derived rational integer-order transfer function, which approximates Eq. (7.2), will have the following form:

$$C_{approx}(s) = \frac{B_m s^m + B_{m-1} s^{m-1} + \ldots + B_1 s + B_0}{A_n s^n + A_{n-1} s^{n-1} + \ldots + A_1 s + A_0}, \tag{7.3}$$

with [m/n] being the order of the numerator and denominator, respectively, of the applied approximation and A_i $(i = 0 \ldots n)$, with B_j $(j = 0 \ldots m)$, being real and positive coefficients.

The derivation of the coefficients A_i and B_j can be performed through the employment of two tools, which are based on MATLAB software: the first one on the built-in functions *frd* and *fitfrd* and the second one on the FLOreS tool. The frequencies ω_i, ω_d, ω_t, and ω_l of the controller transfer function in Eq. (7.2) are related to a common frequency ω_c according to the following equations: $\omega_i =$

$0.1\omega_c$, $\omega_d = \omega_c/a$, $\omega_t = a\omega_c$, and $\omega_l = 10\omega_c$. In addition, the specifications of the controller are the following: $K_P = 0.0645$, $\omega_c = 2\pi \cdot 100\,\text{rad/s}$, $a = 5$, $\lambda = \gamma = 1$, and $\mu = 1.2$ [17, 18].

In the case of the employment of the *frd* and *fitfrd* functions, which perform curve fitting approximation, the whole transfer function in Eq. (7.2) is approximated using a set of numerator and denominator orders equal to order [3/4] within the frequency range of interest $f = \left[10^0, 10^4\right]$ Hz. Using the MATLAB code given in the Appendix, the derived function is given by

$$C_{fitfrd}(s) = \frac{1.921 \times 10^4 s^3 + 1.52 \times 10^7 s^2 + 2.076 \times 10^9 s + 8.041 \times 10^{10}}{s^4 + 1.055 \times 10^4 s^3 + 3.076 \times 10^7 s^2 + 1.972 \times 10^{10} s + 1.088 \times 10^9}. \tag{7.4}$$

The FLOreS tool, on the other hand, approximates the fractional-order parts of the function separately and their product forms the final approximated function. This is demonstrated in Fig. 7.2, where only the lead part is fractional with $\mu = 1.2$. Applying a second-order approximation on this part and then inserting the obtained

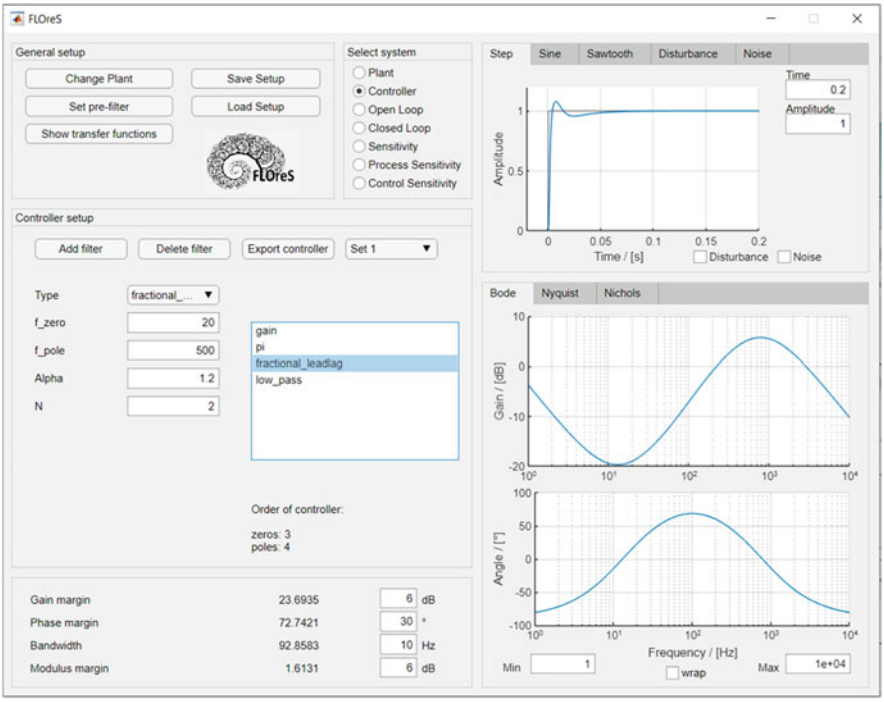


Fig. 7.2 Screenshot of the FLOreS toolbox GUI for approximating the fractional term of the product in Eq. (7.2)

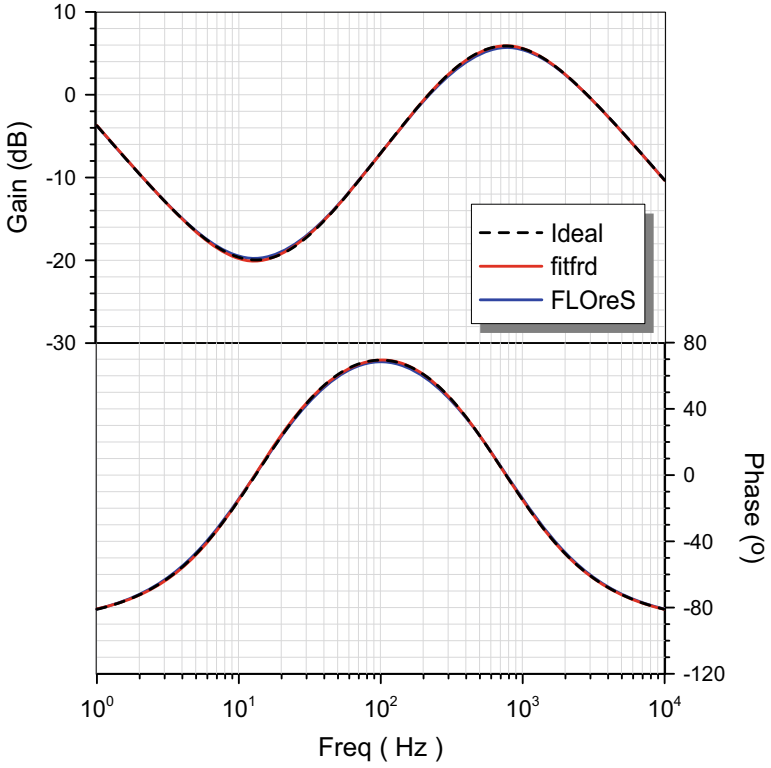


Fig. 7.3 Gain and phase frequency responses of the controller for the *fitfrd* (red solid line) and the FLOreS (blue solid line) approximation methods

function in Eq. (7.2), the derived controller transfer function is given by

$$C_{FLOreS}(s) = \frac{1.929 \times 10^4 s^3 + 1.359 \times 10^7 s^2 + 1.882 \times 10^9 s + 6.936 \times 10^{10}}{s^4 + 1.071 \times 10^4 s^3 + 3.055 \times 10^7 s^2 + 1.711 \times 10^{10} s}. \tag{7.5}$$

The open-loop gain and phase responses for the *fitfrd* (red solid line) and the FLOreS (blue solid line) approximation methods, along with the ideal responses represented by black dashed lines, are depicted in Fig. 7.3. The associated error plots are provided in Fig. 7.4, where the maximum observed errors of both gain and phase are about 1.7% for the *fitfrd* method and 3.1% for the FLOreS method. The most important frequency-response characteristics, including gain margin (G_m), crossover gain frequency (f_{cg}), phase margin (P_m), and phase crossover frequency (f_{cp}), are summarized in Table 7.1.

The closed-loop behavior is evaluated through the step responses given in Fig. 7.5, while the corresponding performance characteristics are provided in

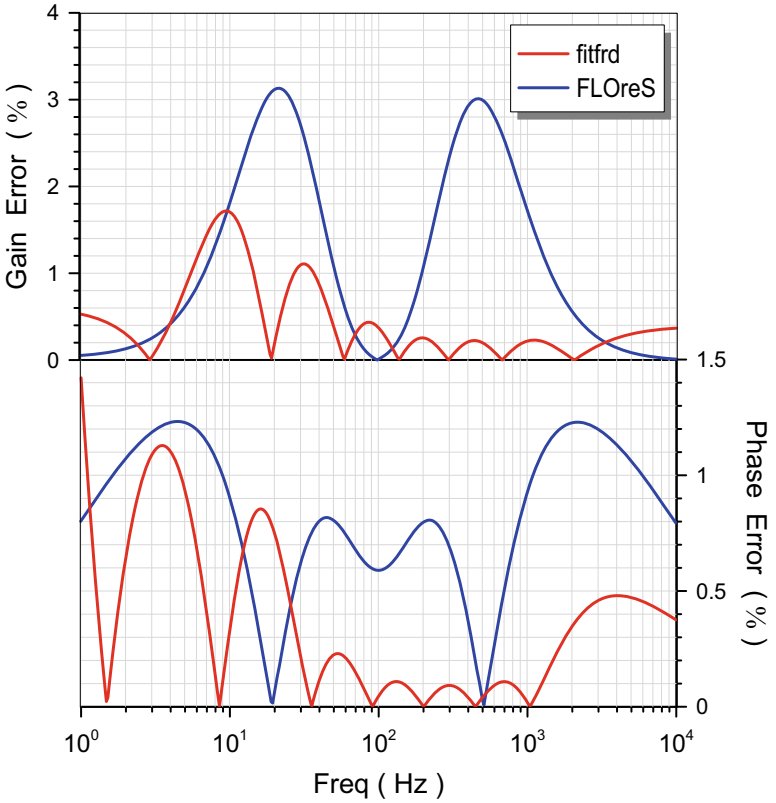


Fig. 7.4 Error plots of the gain and phase frequency responses of the controller for the *fitfrd* and FLOreS approximation methods

Table 7.1 Values of characteristic parameters of the open-loop system for the *fitfrd* and FLOreS approximation methods

Parameter	Ideal	*fitfrd*	FLOreS
G_m (dB)	23.3	23.32	23.7
f_{cg} (Hz)	770.3	771.2	778.5
P_m (°)	73.75	73.77	72.74
f_{cp} (Hz)	92.8	92.4	92.8

Table 7.2. The obtained comparison results show that both tools under consideration offer almost the same level of accuracy.

7.3 Implementation Issues

The implementation of the approximated functions in Eqs. (7.4) and (7.5) can be performed using the partial fraction expansion-based configuration [19] or multi-

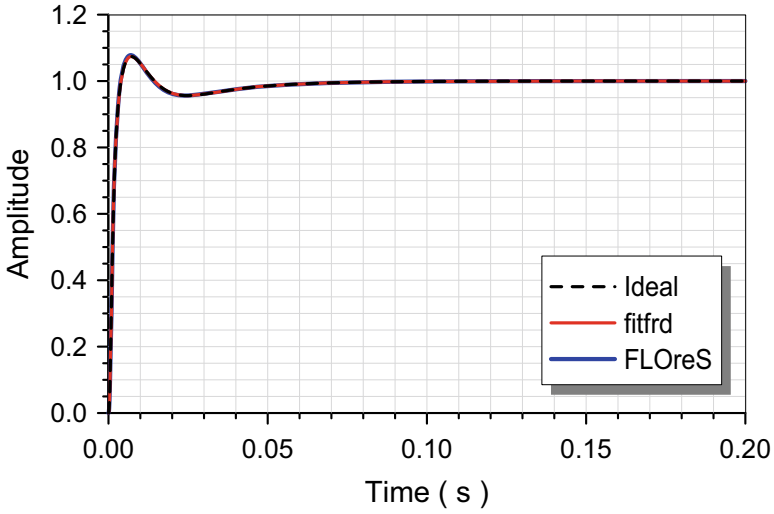


Fig. 7.5 Step response of the closed-loop system controller–plant for the *fitfrd* (red solid line) and the FLOreS (blue solid line) approximation methods

Table 7.2 Values of characteristic parameters of the step response of the closed-loop system for the *fitfrd* and FLOreS approximation methods

Parameter	Ideal	*fitfrd*	FLOreS
Rise time (ms)	2.3	2.3	2.3
Settling time (ms)	44.2	44.3	44.4
Overshoot (%)	7.39	7.47	7.79

feedback structures such Follow-the-Leader-Feedback (FLF) and Inverse Follow-the-Leader-Feedback (IFLF) [20]. The functional block diagrams for the *fitfrd* case are shown in Fig. 7.6.

In order to implement the FLOreS function, it is, initially, formed as

$$C_{FLOreS}(s) = \frac{1}{s} \cdot \frac{1.929 \times 10^4 s^3 + 1.359 \times 10^7 s^2 + 1.882 \times 10^9 s + 6.936 \times 10^{10}}{s^3 + 1.071 \times 10^4 s^2 + 3.055 \times 10^7 s + 1.711 \times 10^{10}}, \tag{7.6}$$

which means that an extra stage (i.e., an integer-order integrator) is required. Then, the rest part is a third-order transfer function, which is implemented by the typical partial fraction expansion-based and multi-feedback structures, as it is shown in Fig. 7.7.

The obtained coefficients for both implementations applying the *fitfrd* and FLOreS approximation methods are given in Tables 7.3 and 7.4, respectively.

Both approximation methods are efficient and lead to a rational integer-order transfer function. For the exact controller specifications, both methods require

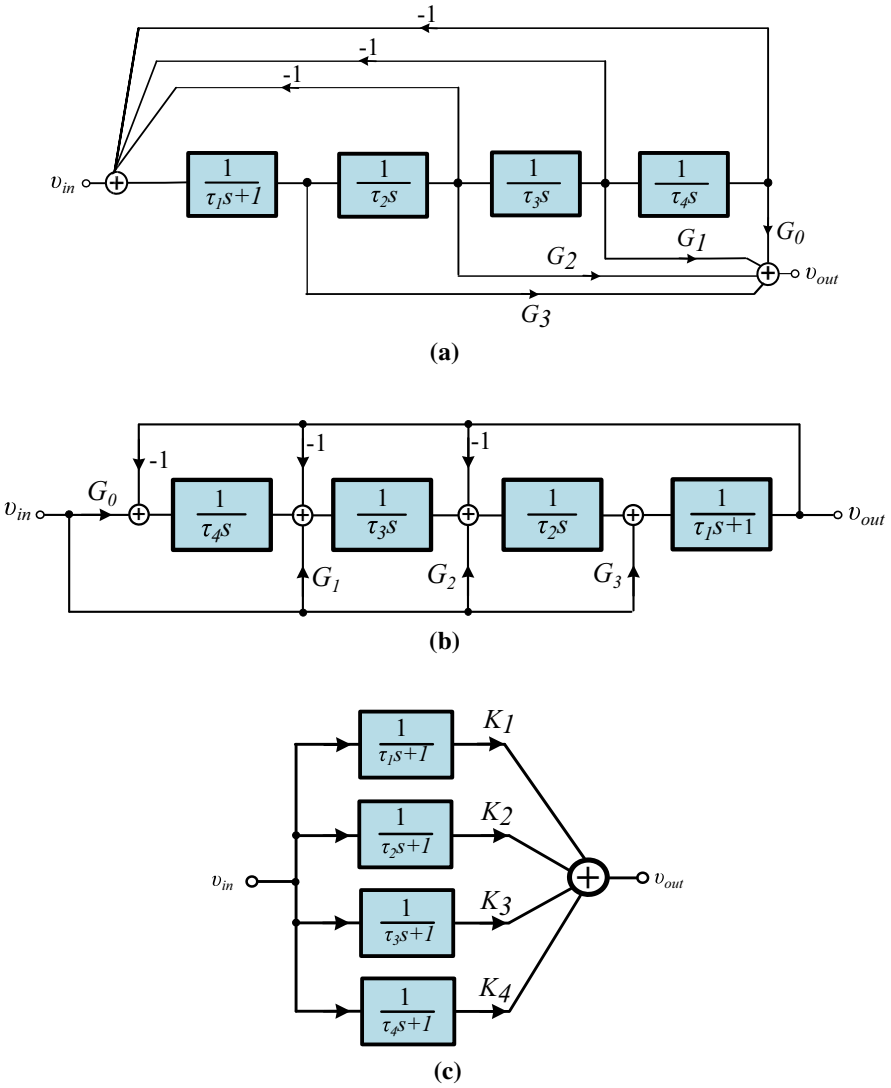


Fig. 7.6 Functional block diagrams based on (**a**) FLF, (**b**) IFLF, and (**c**) PFE schemes for realizing the transfer function in Eq. (7.4), derived using the *fitfrd* approximation method

a fourth-order approximation, with the order of the numerator equal to 3, but of different form. The implementation of the *fitfrd* case, following the partial fraction expansion technique, offers the advantage of absence of the sensitive factor K_0, while in the multi-feedback realization the gain factor and, as a result, a summation stage are omitted leading to a more compact structure. On the other hand, the FLOreS case is implemented by the cascade connection of a third-order

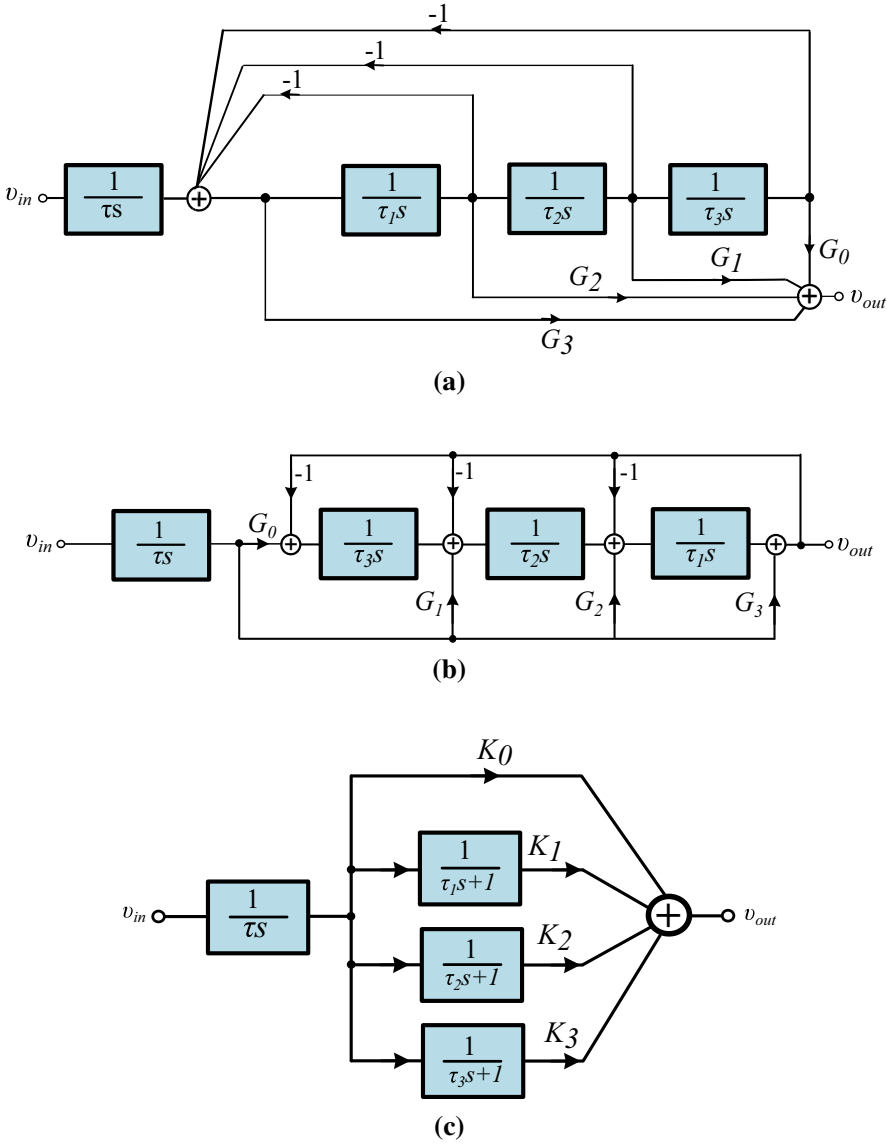


Fig. 7.7 Functional block diagrams based on (**a**) FLF, (**b**) IFLF, and (**c**) PFE schemes for realizing the transfer function in Eq. (7.5), which is derived through the employment of the FLOreS toolbox

partial fraction expansion-based or multi-feedback structure and an integer-order integrator. Considering the scaling factors in Table 7.4, it is obvious that their large values lead to nonpractical specifications for the implementation. In order to avoid this problematical case, an appropriate scaling of the factors to lower values through the time constant of the integer-order integrator can be performed.

Table 7.3 Values of scaling factors and time constants for implementing the controller using the structures in Fig. 7.6a–c and applying the *fitfrd* approximation method

Scaling factors			Time constants		
Variable	Fig. 7.6a,b	Fig. 7.6c	Variable	Fig. 7.6a,b	Fig. 7.6c
K_0/G_0	73.9	–			
K_1/G_1	0.105	9.09	τ_1 (μs)	947.8	169.2
K_2/G_2	0.494	–9.28	τ_2 (μs)	343	266.7
K_3/G_3	1.82	0.288	τ_3 (ms)	1.6	1.1
K_4/G_4	–	73.82	τ_4 (s)	18.13	18.13

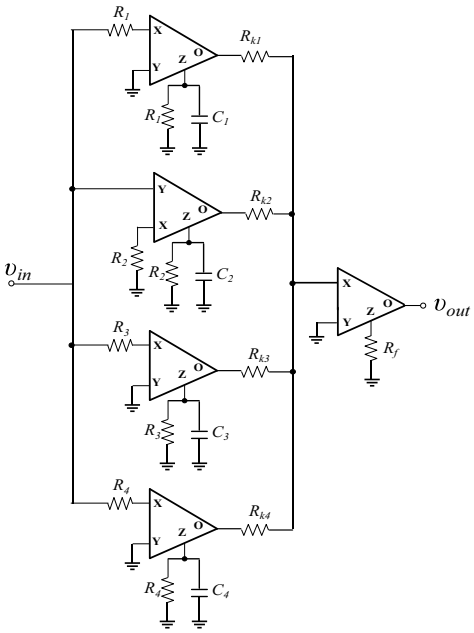
Table 7.4 Values of scaling factors and time constants for implementing the controller using the structures in Figs. 7.7a–c and applying the FLOreS approximation method

Scaling factors			Time constants		
Variable	Fig. 7.7a,b	Fig. 7.7c	Variable	Fig. 7.7a,b	Fig. 7.7c
K_0/G_0	4.054	19,290			
K_1/G_1	61.6	−47,290	τ_1 (μs)	93.37	159.3
K_2/G_2	1268.9	28,142	τ_2 (μs)	350.6	270.7
K_3/G_3	19,290	−138	τ_3 (ms)	1.8	1.4
K_4/G_4	–	–	τ_4 (s)	–	–

In order to have a comparison in terms of hardware of the implementations derived from Figs. 7.6c and 7.7c, and using Current Feedback Operational Amplifiers (CFOAs) as active elements, the resulting structures are demonstrated in Figs. 7.8a,b. As the number of the required CFOAs, resistors, and capacitors in the topology of Fig. 7.8a is 5, 13, and 4, respectively, while for the topology in Fig. 7.8b is 6, 14, and 4, respectively, it is obvious that the topology derived through the utilization of the *fitfrd* tool is simpler. It must be also mentioned at this point that the corresponding operational amplifier (op-amp)-based implementation presented in [4] is performed using 6 op-amps, 15 resistors, and 4 capacitors, proving the superiority of the structure presented in this manuscript.

7.4 Simulation Results

Using the values provided in Table 7.3 and the fact that the implemented time constants are given by the formula $\tau_i = R_i C_i$ (i=1,2,3,4), and then assuming that $R = R_f = 10\,\text{k}\Omega$ and $R_1 = R_2 = R_3 = R_4 = 20\,\text{k}\Omega$, the values of capacitors, rounded to the E96 series defined in IEC 60063, were $C_1 = 1.69\,\text{nF}$, $C_2 = 2.67\,\text{nF}$, $C_3 = 11\,\text{nF}$, and $C_4 = 182\,\mu\text{F}$. Employing the AD844 discrete component ICs as CFOAs in the OrCAD PSpice simulator, with DC power supply voltages equal to $\pm 15\,\text{V}$, the obtained frequency responses of the controller, along with the theoretically predicted plots (black dashed lines), are demonstrated in Fig. 7.9, along with the gain and phase errors.



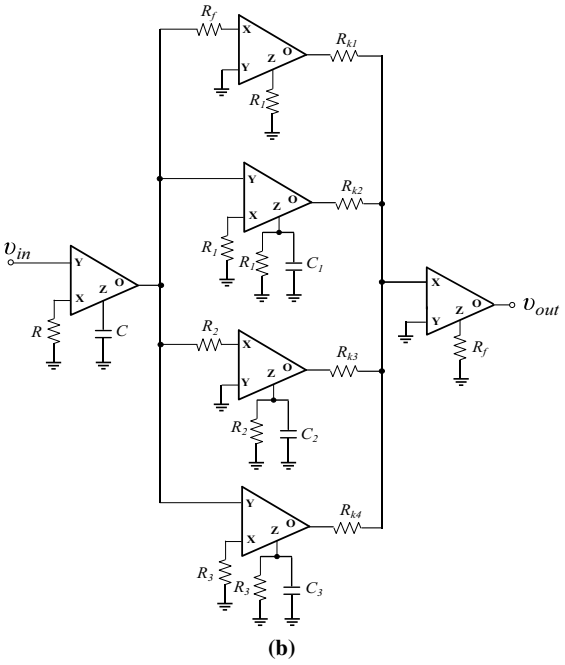


Fig. 7.8 CFOA-based implementations of the PFE-based block diagrams in (**a**) Fig. 7.6c (using the *fitfrd* tool) and (**b**) Fig. 7.7c (using the FLOreS toolbox)

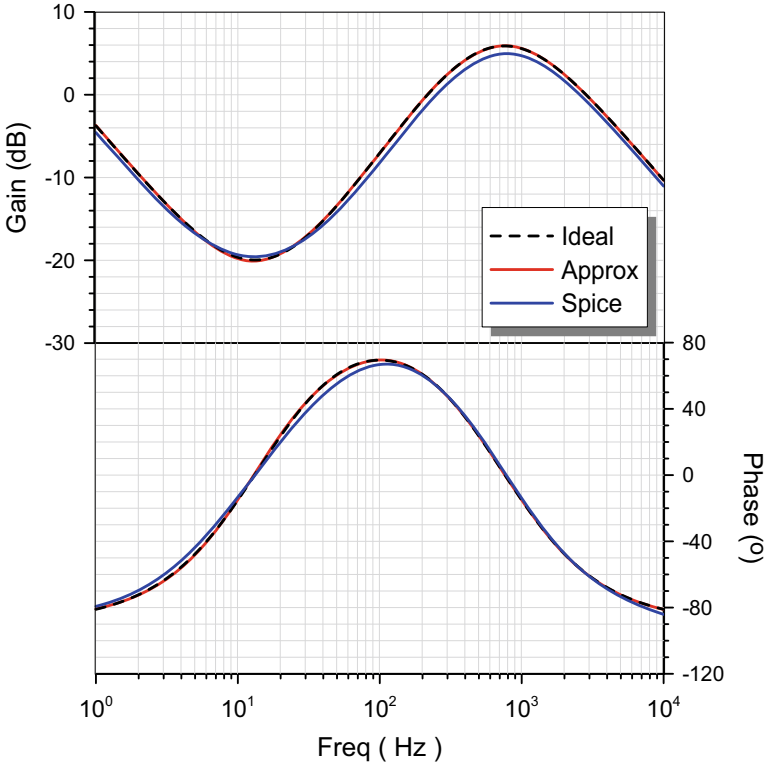


Fig. 7.9 Simulated gain and phase frequency responses of the controller

The functionality of the open-loop system controller–plant is verified through the frequency responses of Fig. 7.10, and the most important frequency-response characteristics, including gain margin (G_m), crossover gain frequency (f_{cg}), phase margin (P_m), and phase crossover frequency (f_{cp}), are summarized in Table 7.5.

The closed-loop system behavior is evaluated through the step responses given in Fig. 7.11, while the corresponding performance characteristics are provided in Table 7.6.

The sensitivity performance of the system is evaluated, assuming a 2% maximum tolerance of the values of passive elements. The obtained standard deviation and mean values of the critical characteristics regarding the open-loop system (i.e., gain margin, crossover gain frequency, phase margin, and phase crossover frequency) and the closed-loop system (i.e., rise time, settling time, and overshoot) for $N = 500$ runs are summarized in Table 7.7, while, indicatively, the statistical histograms for the gain margin, phase margin, and settling time are presented in Fig. 7.12 These results confirm the reasonable sensitivity characteristics of the system.

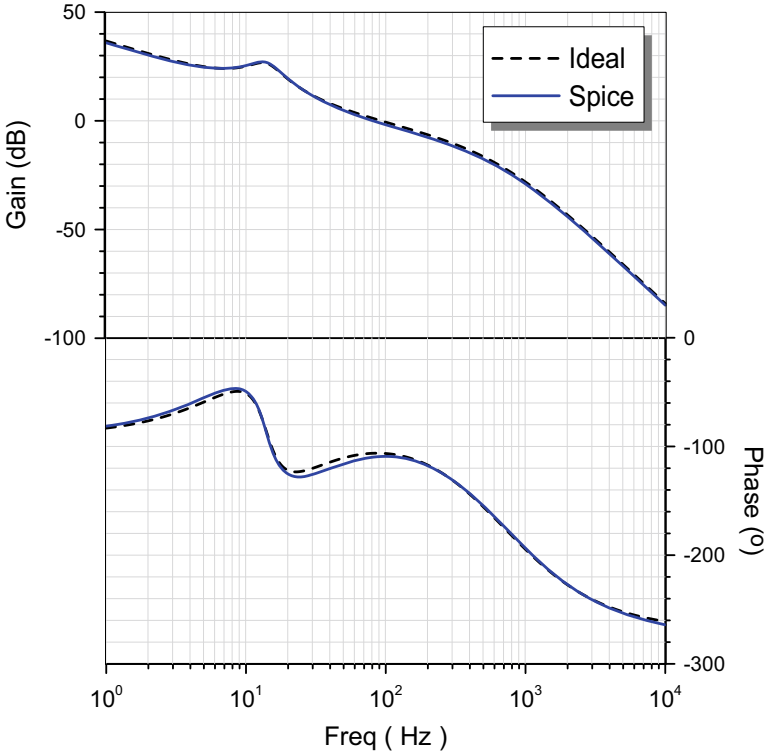


Fig. 7.10 Simulated open-loop gain and phase frequency responses of the controller–plant system

Table 7.5 Simulated values of the critical characteristics of the open-loop controller–plant system

Parameter	Ideal	Simulation
G_m (dB)	23.3	24.6
f_{cg} (Hz)	770.3	787.2
P_m (°)	73.75	70.14
f_{cp} (Hz)	92.8	81.1

7.5 Conclusions

In this work, a comparison between two approximation tools is performed, considering a fractional-order control system suitable for a three-degree-of-freedom planar precision positioning mechatronic system. The first step was the evaluation of the accuracy of each tool, with the obtained results showing that it is on the same level in both cases. The realization of the resulting integer-order rational transfer functions leads to topologies with different complexity. The reduced order of the numerator of the transfer function in the case of the employment of the *fitfrd* function is advantageous in terms of active and passive component count, as well as of the spread of passive elements' values. On the other hand, FLOreS toolbox offers

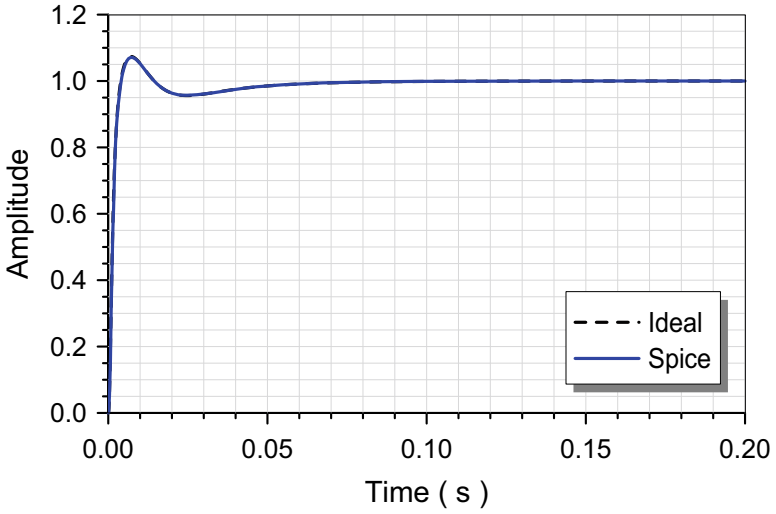


Fig. 7.11 Step response of the closed-loop system controller–plant

Table 7.6 Simulated values of the characteristic parameters of the step response of the closed-loop system

Parameter	Ideal	Simulation
Rise time (ms)	2.3	2.5
Settling time (ms)	44.2	46.5
Overshoot (%)	7.39	10.3

Table 7.7 Statistical values of the critical characteristics of the open-loop and closed-loop controller–plant system

Parameter	Mean	Std Dev
G_m (dB)	24.7 (theor. 23.3)	0.35
f_{cg} (Hz)	787.6 (theor. 770.3)	6.4
P_m (°)	69.95 (theor. 73.75)	0.87
f_{cp} (Hz)	80.8 (theor. 92.8)	3.6
Rise time (ms)	2.55 (theor. 2.3)	0.09
Settling time (ms)	46.5 (theor. 44.2)	0.59
Overshoot (%)	10.4 (theor. 7.39)	0.9

a user-friendly interface with several capabilities including tuning, evaluation of open-loop and closed-loop system behavior, sensitivity, noise performance, etc. The provided simulation results confirm the aforementioned derivations. Future research steps include the application of these tools to controller functions with more than one fractional part, where it is expected that the advantage of the curve fitting-based approximation (*fitfrd*) over the FLOreS toolbox will be more evident, owing to the fact that in this case the transfer function is approximated as a whole and not through the approximation of each part separately.

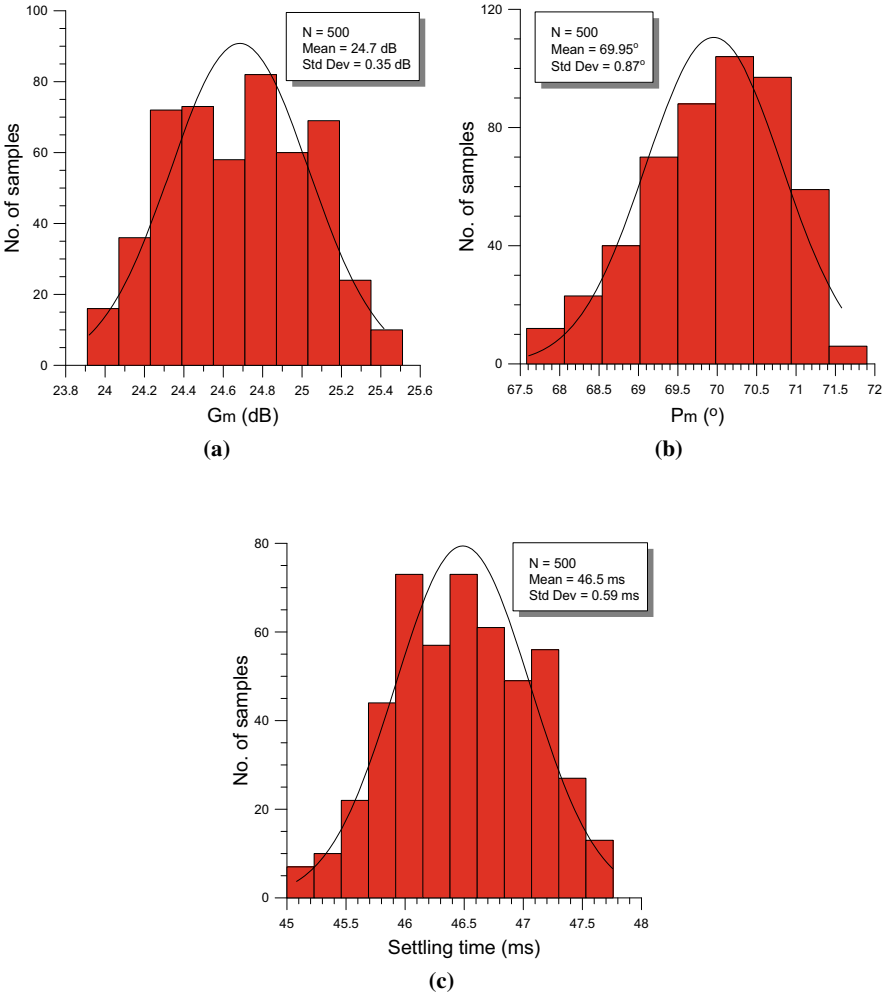


Fig. 7.12 Monte Carlo statistical histograms for the (**a**) gain margin, (**b**) phase margin, and (**c**) settling time of the system

Appendix

```
%% Charactersitics of the Controller
Kp=0.0645;
fc = 100;
wc=2*pi*fc;
alpha=5;
```

```
wi=0.1*wc;
wd=wc/alpha;
wt=wc*alpha;
wl=10*wc;
lamda=1;
mu=1.2;
gamma=1;
% Frequency range of interest
fmin = 1E+0; fmax = 1E+4;
freq = logspace(log10(fmin),log10(fmax),201);
wmin = 2*pi*(fmin); wmax = 2*pi*(fmax);
w = logspace(log10(wmin),log10(wmax),201);
% approximation order
m = 3;
n = 4;

%% Controller Transfer Function
% Intermediate transfer functions
s = tf('s');
tf1_inter = (1+wi/s);
tf2_inter = (1+s/wd)/(1+s/wt);
tf3_inter = 1/(1+s/wl);
tf1_resp = (freqresp(tf1_inter,w)).^lamda;
tf2_resp = (freqresp(tf2_inter,w)).^mu;
tf3_resp = (freqresp(tf3_inter,w)).^gamma;
% Total transfer function
C_resp = Kp*tf1_resp.*tf2_resp.*tf3_resp;
C_resp_data = frd(C_resp,w);

%% Approximation of the Controller Transfer Function
% fitfrd MATLAB built-in function
C_fitfrd = fitfrd(C_resp_data,n,n-m);
C_fitfrd = tf(C_fitfrd)
```

References

1. D. Xue, Y. Chen, Fractional order calculus and its applications in mechatronic system controls organizers, in *2006 International Conference on Mechatronics and Automation* (IEEE, 2006), pp. nil33–nil33
2. C.I. Muresan, C. Copot, C. Ionescu, R. De Keyser, Robust fractional order control of LPV dynamic mechatronic systems, in *2019 IEEE 15th International Conference on Control and Automation (ICCA)* (IEEE, 2019), pp. 154–9
3. C. Copot, C.M. Ionescu, C.I. Muresan, *Image-Based and Fractional-Order Control for Mechatronic Systems: Theory and Applications with MATLAB®* (Springer Nature, 2020)

4. S. Kapoulea, C. Psychalinos, A.S. Elwakil, S.H. HosseinNia, Realizations of fractional-order PID loop-shaping controller for mechatronic applications. Integration **80**, 5–12 (2021)
5. L. Bruzzone, M. Baggetta, P. Fanghella, Fractional-order $PII^{1/2}DD^{1/2}$ control: Theoretical aspects and application to a mechatronic axis. Appl. Sci. **11**(8), 3631 (2021)
6. I. Podlubny, I. Petraš, B.M. Vinagre, P. O'leary, L. Dorčák, Analogue realizations of fractional-order controllers. Nonlinear Dyn. **29**(1), 281–96 (2002)
7. C.A. Monje, Y. Chen, B.M. Vinagre, D. Xue, V. Feliu-Batlle, *Fractional-order Systems and Controls: Fundamentals and Applications* (Springer Science & Business Media, 2010)
8. C. Muñiz-Montero, L.V. García-Jiménez, L.A. Sánchez-Gaspariano, C. Sánchez-López, V.R. González-Díaz, E. Tlelo-Cuautle, New alternatives for analog implementation of fractional-order integrators, differentiators and PID controllers based on integer-order integrators. Nonlinear Dyn. **90**(1), 241–256 (2017)
9. E. Tlelo-Cuautle, A.D. Pano-Azucena, O. Guillén-Fernández, A. Silva-Juárez, *Analog/digital Implementation of Fractional Order Chaotic Circuits and Applications* (Springer, 2020)
10. A.G. Soriano-Sánchez, J. Soto-Vega, E. Tlelo-Cuautle, M.A. Rodríguez-Licea, Fractional-order approximation of PID controller for buck–boost converters. Micromachines **12**(6), 591 (2021)
11. A. Tepljakov, B.B. Alagoz, C. Yeroglu, E.A. Gonzalez, S.H. Hosseinnia, E. Petlenkov, A. Ates, M. Cech, Towards industrialization of FOPID controllers: A survey on milestones of fractional-order control and pathways for future developments. IEEE Access **9**, 21016–42 (2021)
12. M.A. George, D.V. Kamat, C.P. Kurian, Electronically tunable ACO based fuzzy FOPID controller for effective speed control of electric vehicle. IEEE Access **9**, 73392–412 (2021). https://doi.org/10.1109/ACCESS.2021.3080086
13. S. Kapoulea, C. Psychalinos, A.S. Elwakil, Realization of Cole–Davidson function-based impedance models: Application on plant tissues. Fractal Fract. **4**(4), 54 (2020)
14. G. Tsirimokou, C. Psychalinos, Ultra-low voltage fractional-order circuits using current mirrors. Int. J. Circuit Theory Appl. **44**(1), 109–126 (2016)
15. S. Kapoulea, C. Psychalinos, A.S. Elwakil, Power law filters: A new class of fractional-order filters without a fractional-order Laplacian operator. AEU Int. J. Electron. Commun. **129**, 153537 (2021)
16. L. van Duist, G. van der Gugten, D. Toten, N. Saikumar, H. HosseinNia, FLOreS-fractional order loop shaping MATLAB toolbox. IFAC-PapersOnLine **51**(4), 545–50 (2018)
17. S.H. HosseinNia, N. Saikumar, Fractional-order precision motion control for mechatronic applications, in *Handbook of Fractional Calculus with Applications*, vol. 6 (Walter de Gruyter GmbH & Co KG, 2019), pp. 339–356
18. N. Saikumar, R.K. Sinha, S.H. HosseinNia, 'Constant in gain lead in phase' element-Application in precision motion control. IEEE/ASME Trans. Mechatron. **24**(3), 1176–1185 (2019)
19. P. Bertsias, C. Psychalinos, B.J. Maundy, A.S. Elwakil, A.G. Radwan, Partial fraction expansion–based realizations of fractional-order differentiators and integrators using active filters. Int. J. Circuit Theory Appl. **47**(4), 513–531 (2019)
20. G. Tsirimokou, C. Psychalinos, A. Elwakil, *Design of CMOS Analog Integrated Fractional-order Circuits: Applications in Medicine and Biology* (Springer, 2017)

Part IV
Chaotic Systems

Chapter 8
Comparative Analysis of Chaotic Features of Maps Without Fixed Points

Claudio García-Grimaldo and Eric Campos-Cantón

Abstract In the second decade of this century, chaotic maps without fixed points, which contain hidden attractors, have been described. The chaotic properties that exist between them and with maps with fixed points may differ. In this manuscript, we will make a comparative analysis of some of these features between discrete maps with and without fixed points, both one-dimensional and two-dimensional. The analyses performed on the maps will be by means of bifurcation diagrams, Lyapunov exponents, entropy calculation, as well as other statistical and numerical tools. Finally, tables are used to show and compare the values obtained from each of the exposed maps.

Keywords Dynamical system · Discrete map · Map without fixed points · Chaotic map · Lyapunov exponent · Bifurcation diagram

8.1 Introduction

The importance of chaotic dynamical systems in different areas of science, engineering, and technological applications is well known [1, 2]. Both continuous and discrete time dynamical systems have been widely studied mainly to know their topological [3, 4], dynamical, and ergodic properties [5, 6], as well as for their use in concrete applications [1, 7, 8]. Many of the works reported before the twenty-first century employ dynamical systems in which fixed points exist [9]. These fixed points help to identify the dynamic behavior around them. For example, Shil'nikov's theorem allows us to identify chaotic behavior in the neighborhoods of the parameter space where certain homoclinic orbits surrounding the saddle-focus equilibrium point appear. The chaotic behavior generated by the Shil'nikov method is called as homoclinic chaos or heteroclinic chaos [10].

C. García-Grimaldo (✉) · E. Campos-Cantón
Instituto Potosino de Investigación Científica y Tecnológica A.C., San Luis Potosi, Mexico
e-mail: claudio.garcia@ipicyt.edu.mx; eric.campos@ipicyt.edu.mx

G. Huerta Cuellar et al. (eds.), *Complex Systems and Their Applications*,
https://doi.org/10.1007/978-3-031-02472-6_8

In this century, a new classification has emerged from the consideration of the nonexistence of fixed points. The first reported works focused on continuous chaotic dynamical systems [11–13]; however, in recent years, systems in discrete time that exhibit chaos have also been reported [14–18]. In both cases and given the absence of fixed points, the analysis of their dynamics is carried out through other methods such as phase portraits, Lyapunov exponents, bifurcation diagrams, period doubling route to chaos, Poincaré maps, and entropy analysis. Dynamical systems without equilibrium/fixed points are considered within those systems that have hidden attractors since they satisfy the definition of hidden attractors [19]. Generally, hidden attractors in vector fields with equilibria imply multistability where self-exited attractors and hidden attractors coexist. Hidden attractors are important in applications because they allow unexpected responses to disturbances in some structures like a bridge or an airplane wing [20]. The lack of fixed points complicates the analysis of the dynamics of the systems, then, maps without fixed points could have a very plausible application in encryption schemes or for the generation of pseudorandom sequences that are cryptographically secure [21], as well as can also be used in the amplitude control technique [18], which offers a high versatility in chaos applications such as chaotic encryption and secure communications[22–25].

As we have already said, discrete systems with chaotic behavior have been recently reported, and therefore, no comparative analysis has been made between them and those systems in which there are fixed points, so it is convenient to make some comparative studies of their chaotic characteristics to have a wider range of these types of systems, as well as to choose the most appropriate parameters and maps in future applications such as in the generation of pseudorandom sequences. In this chapter, we present some of the existing chaotic maps without fixed points, unidimensional and bidimensional reported in the literature with the intention of making an analysis of their chaotic aspects, together with a comparison between these maps and some dynamical systems with fixed points. The first step was to find existing maps in the literature, both one-dimensional and two-dimensional, with chaotic behavior and without fixed points. From these, we chose those maps in which a similar analysis had been made to determine their chaotic properties, along with those that were defined in equal topological spaces and in which time was a discrete variable and space a continuous one. For the comparative analysis, the following issues will be implemented: cobweb plots, phase space diagrams, bifurcation diagrams, Lyapunov exponents, histograms, time series, and sensibility analysis. In addition, entropy calculation test will be performed. Finally, we establish some comparative tables with the intention of contrasting the differences and similarities between the studied maps.

8.2 Unidimensional Maps

8.2.1 Logistic Map

The first map for which we will study its dynamics is the logistic map [26]. This one-dimensional map is defined as

$$x_{n+1} = \mu x_n(1 - x_n). \tag{8.1}$$

If we differentiate (8.1), then we have

$$x'_{n+1} = \mu(1 - 2x_n). \tag{8.2}$$

If we equate (8.2) to zero, then we have that the maximum value at x_{n+1} is when $x_n = \frac{1}{2}$, for this value, $x_{n+1} = \frac{\mu}{4}$. Therefore, if we take $x_n \in [0, 1]$, then $x_{n+1} \in [0, 1]$ only if $\mu \in [0, 4]$, for all $n \in \mathbb{N}$. If $\mu < 0$, then $x_{n+1} < 0$. If $\mu > 4$, then the maximum value at $x_n = \frac{1}{2}$ is greater than one, thus $x_n + 1 \notin [0, 1]$ and $x_{n+1} \to -\infty$ when $n \to \infty$. From this, the range of control parameter μ is $0 \leq \mu \leq 4$.

Given the quadratic factor in (8.1), the logistic map is defined by a family of parabolas in the closed interval [0, 1]. In Fig. 8.1, we can see the graph of the logistic map when the parameter μ takes distinct values.

Fixed Points The logistic map has two fixed points; for the location of them, the following must be satisfied:

$$\mu x_n^*(1 - x_n^*) = x_n^*, \tag{8.3}$$

which is true for the values $x_1 = 0$ and $x_2 = 1 - \dfrac{1}{\mu}$.

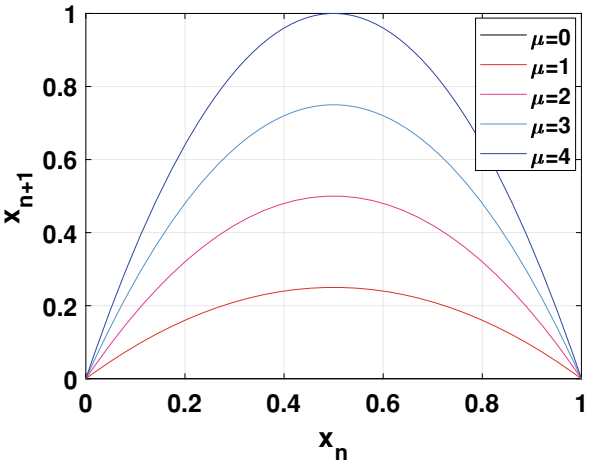
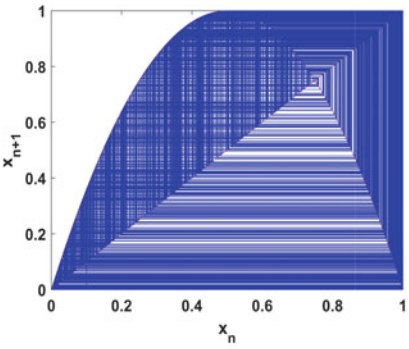
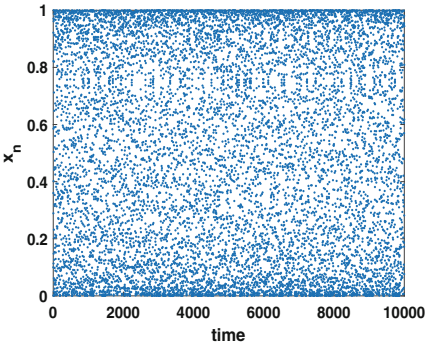


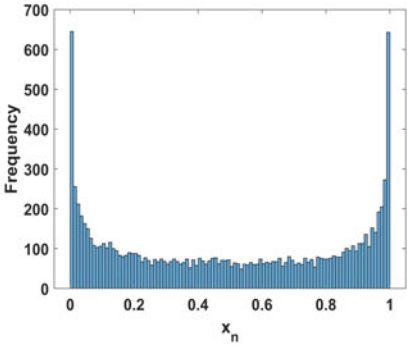
Fig. 8.1 Logistic map with different μ values



(a) Cobweb plot for x_0 and $\mu = 4$.



(b) Time series.



(c) Histogram.

Fig. 8.2 Chaotic dynamics of logistic map. (**a**) Cobweb plot for x_0 and $\mu = 4$. (**b**) Time series. (**c**) Histogram

According to the literature, the logistic model presents chaotic dynamics for certain values of the parameter μ, in particular when $\mu = 4$. In Fig. 8.2, we can observe one of the consequences of the existence of chaos; the non-periodic orbit generated from the initial condition $x_0 = 0.45$ covers a large part of the interval [0,1]. This same behavior can be checked with the time series and histogram, both generated from the given initial condition. In both graphs, it can be observed again that the trajectory occupies a large part of the interval, and however, it can be seen that there are regions where it does so more frequently, i.e., there is no uniform distribution. In the next section, we will show other resources to evaluate the existence of chaos both for this map and for the rest of the one-dimensional maps described in this chapter.

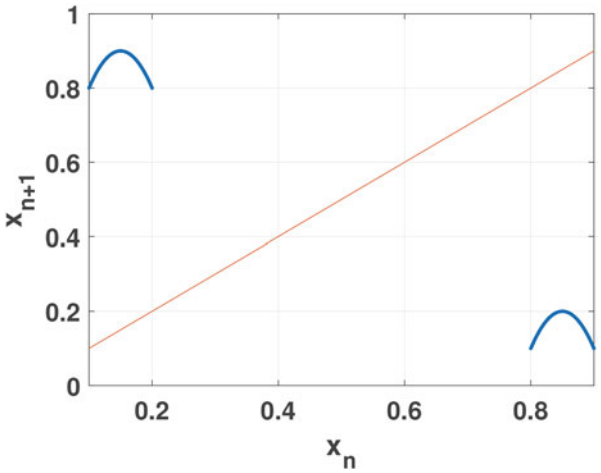


Fig. 8.3 Vertigo-2 map, with parameter $\mu_1 = 40$

8.2.2 Vertigo-2 Map

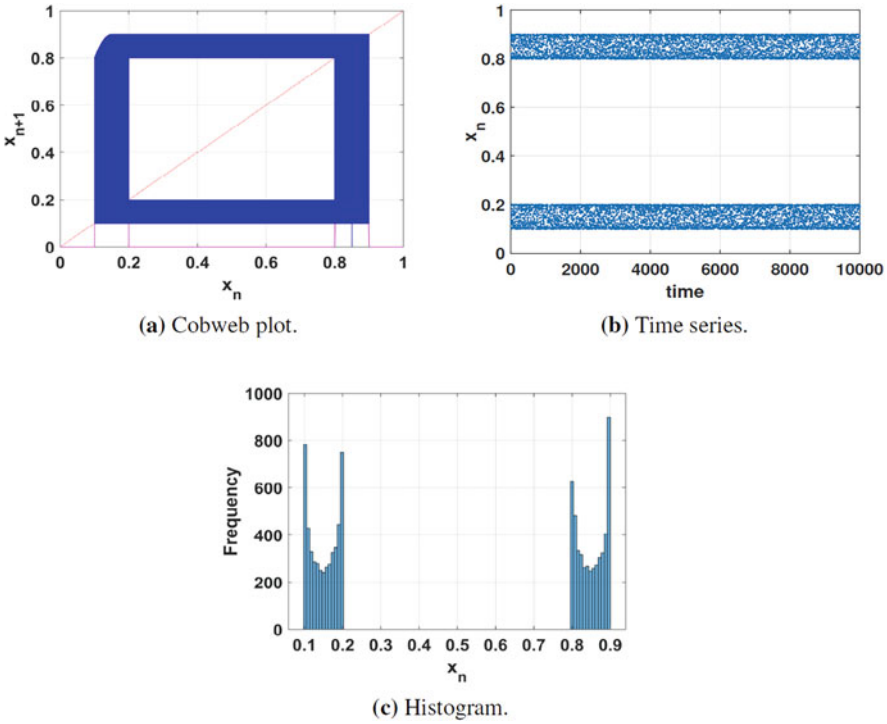
The Vertigo-2 map is a one-dimensional map proposed in [14] and is inspired from logistic map, and its mathematical model is defined as

$$x_{n+1} = \begin{cases} 0.8 + \mu_1(x_n - 0.1)(0.2 - x_n), & \text{if } 0.1 < x_n < 0.2; \\ 0.1 + \mu_1(x_n - 0.8)(0.9 - x_n), & \text{if } 0.8 < x_n < 0.9. \end{cases} \tag{8.4}$$

In this map, the range of control parameter μ_1 is $0 < \mu_1 < 40$. As can be seen in Fig. 8.3, Vertigo-2 map is a type of scaled logistic map that appears in two distinct regions.

Fixed Points This kind of map has no fixed points. It can be verified if we analyze (8.4): for any value of x_{n+1} in the set domain, if $x_n \in (0.1, 0.2)$, then $x_{n+1} \in (0.8, 0.9)$, and if $x_n \in (0.8, 0.9)$, then $x_{n+1} \in (0.1, 0.2)$. In any case, $x_n \neq x_{n+1}$. When the basin of attraction of any attractor does not intersect with any neighborhood of the fixed point, then we have a hidden attractor. Since in this case there are no fixed points, then, the attractors generated will be of this type. In Fig. 8.3, we can see that for Vertigo-2 map, the range is a subset of its domain, as a consequence all trajectories are bounded between the intervals $(0.1, 0.2)$ and $(0.8, 0.9)$, and then, the basin of attraction is a hidden attractor that belongs to these intervals.

In Fig. 8.4a, we can see the trajectory of the orbit when $\mu_1 = 40$ and the initial condition $x_0 = 0.45$. In a similar way to logistic map, the orbit of Vertigo-2 map covers most of its domain, and moreover, at each iteration, it alternates between the first domain and the second domain. In Fig. 8.4b and c, the time series



(a) Cobweb plot. (b) Time series.

(c) Histogram.

Fig. 8.4 Non-periodic dynamics of one orbit for Vertigo-2 map. (**a**) Cobweb plot. (**b**) Time series. (**c**) Histogram

and the histogram allow us to observe this particularity from another perspective. Figure 8.4c shows a histogram similar to the one for the logistic map in each of the sub-domains established for the Vertigo-2 map.

8.2.3 PWL Map

The third kind of unidimensional map presented here is a piecewise linear map (PWL), which is proposed and analyzed in [18]. This one-dimensional map is defined as

$$x_{n+1} = \begin{cases} m_1 x_n + b_1, \text{ if } x_n \leq -a < 0; \\ m_2 x_n + b_2, \text{ if } -a < x_n < 0; \\ m_2 x_n - b_2, \text{ if } \ 0 \leq x_n < a; \\ m_1 x_n - b_1, \text{ if } \quad x_n \geq a, \end{cases} \tag{8.5}$$

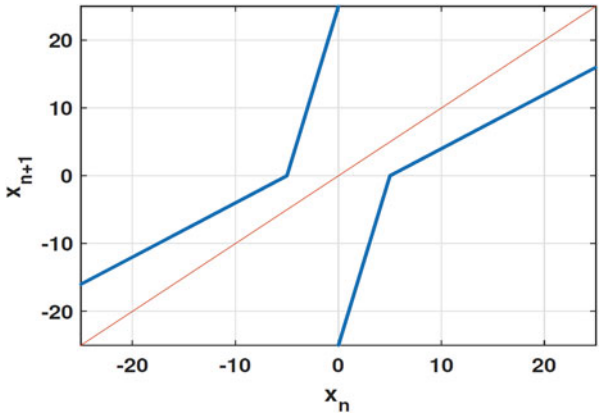


Fig. 8.5 PWL map given by (8.5). $m_1 = 0.8$, $m_2 = 5$, and $b_1 = 4$

where $m_1 \neq 0$ and $m_2 \neq 0$. The values a and b_2 are adjusted according to equations

$$a = \frac{b_1}{m_1}, \tag{8.6}$$

$$b_2 = \frac{m_2\, b_1}{m_1}. \tag{8.7}$$

From this, in [18], the following proposition is stipulated.

Proposition 1 *If $m_i > 0$, then $b_i > 0$, and on the other hand, if $m_i < 0$, then $b_i < 0$, with $i = 1, 2$.*

Fixed Points The PWL map defined in (8.5) under certain values of its parameters has no fixed points. In [18], the following theorem is established, in which the conditions for the absence of such points are given.

Theorem 1 *Let f be a class of PWL maps defined by (8.5), with $m_1 \in (-\infty, 0) \cup (0, 1)$, and $m_2 \in (0, \infty)$, then the class of PWL maps does not have fixed points in $\mathbb{R}$.*

An example of this kind of map can be seen in Fig. 8.5. Figure 8.6a shows that in the graph of the cobweb diagram, the orbit covers a large part of a bounded region, and the time series and the histogram in Fig. 8.6b and c, respectively, highlight this phenomenon, i.e., non-periodic fluctuations are visualized over time within an interval. In this case, the basin of attraction belongs to the interval $[-25, 25]$, and then the hidden attractor is generated into this region.

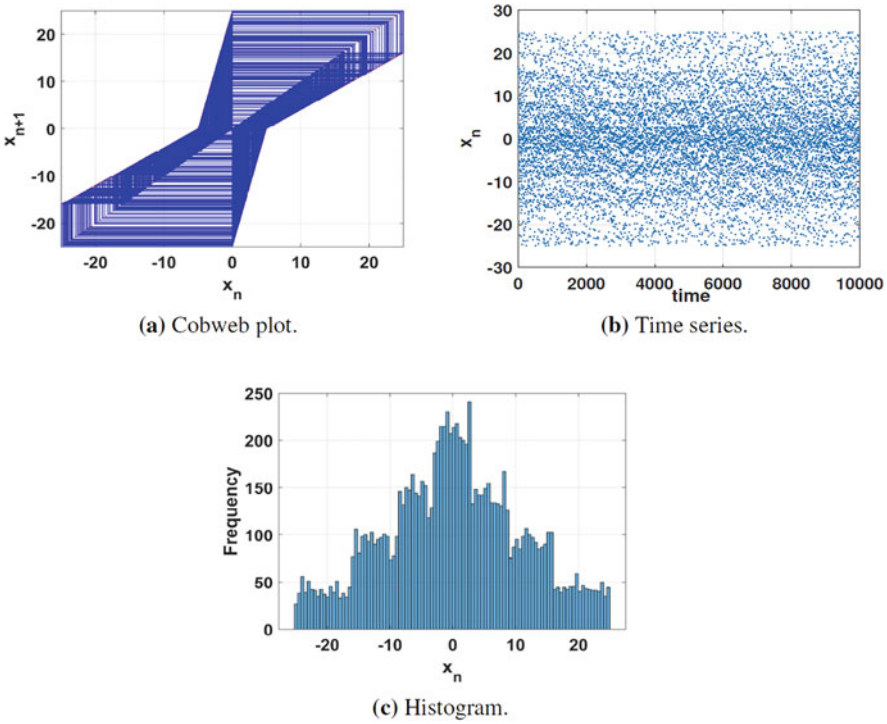


Fig. 8.6 Non-periodic dynamics of PWL map: $m_1 = 0.8$, $m_2 = 5$, $b_1 = 4$, and $x_0 = 0.45$. (**a**) Cobweb plot. (**b**) Time series. (**c**) Histogram

8.2.4 Bifurcation Diagrams

In this section, we will carry out an analysis of the dynamics for each of the three maps described in the previous section. One way to check if there are other values for which there is non-periodic behavior is through the implementation of bifurcation diagrams.

Figure 8.7a exhibits the bifurcation diagram of logistic map when the parameter μ is varied in the interval $(0, 1)$, with an initial condition $x_0 = 0.1$. Here we can see that there are ranges where the orbit is periodic and ranges where it is not. In addition, we can detect that at $\mu = 3$, logistic map presents a period-doubling bifurcation. Then, at $\mu \approx 3.57$, the system enters into a chaotic state and continues to manifest itself until the value of $\mu = 4$ (in certain regions between these two values, there are interior crisis bifurcations, where chaotic dynamics ceases to manifest itself). Figure 8.7b depicts the bifurcation diagram of Vertigo-2 map when the initial condition is 0.1. As can be seen, and given that this map has no fixed points, it starts with a double period. Moreover, the geometric structure of the bifurcation diagram is a scale of the one generated for the logistic map

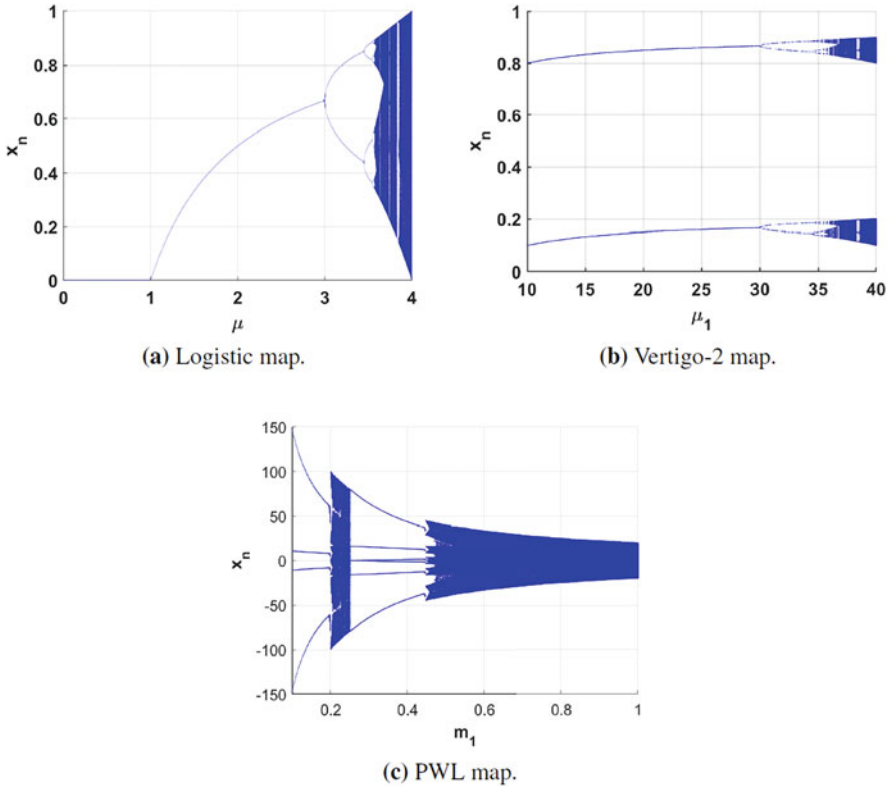


Fig. 8.7 Bifurcation diagrams. (**a**) Logistic map. (**b**) Vertigo-2 map. (**c**) PWL map

and therefore inherits certain properties of the diagram of this last map, such as period-doubling bifurcations, interior crisis bifurcations and non-periodic dynamics. Figure 8.7c shows the bifurcation diagram for PWL map, with $m_1 \in [0.1, 1]$, $m_2 = 5$, $b_1 = 4$, and an initial condition $x_0 = 0.45$. In general, it is possible to observe two behaviors: periodic and chaotic. The periodic behavior appears in two windows: the first of period four in the interval $(0.1, 0.2)$ and the second of period six when $0.25 < m_1 < 0.45$. Furthermore, there are two windows of chaotic behavior: $0.2 < m_1 < 0.25$ and $0.45 < m_1 \leq 1$. Note that in contrast to the Vertigo-2 map, this does not present a doubling periodic bifurcation, but rather unexpectedly enters into a chaotic state for certain values and again abruptly produces periodic dynamics to finally emerge again in a chaotic condition.

Table 8.1 shows the characteristics described above for the three aforementioned maps.

Table 8.1 Features chaotics and periodics for the maps

Map	First period	Bifurcation	Periodic behavior	Chaos
Logistic map	1	Periodic-doubling interior crisis	(0, 3.7)	(3.7, 4]
Vertigo-2 map	2	Periodic-doubling interior crisis	(0, 37)	(37, 40]
PWL map	4	Crisis bifurcation	(0, 0.2) ∪ (0.25, 0.45)	(0.2, 0.25) ∪ (0.45, 0, 1]

8.2.5 Lyapunov Exponents

There are several definitions that can be used to determine chaos, for example, the numerical calculation of Lyapunov exponents allows us to determine the regions where chaos occurs under the condition that the orbits are bounded and present a positive Lyapunov exponent [27]. In this section, we show the existence of chaotic behavior for the three unidimensional maps through Lyapunov exponents. The calculation of the Lyapunov exponent is carried out as

$$\lambda(x_0) = \lim_{n\to\infty} \frac{1}{n} \sum_{k=0}^{n-1} \ln \mid f'(x(k)) \mid. \tag{8.8}$$

First, and as seen above, for logistic map, there are zones with non-periodic behavior. Thus, by using the Lyapunov exponents, we can verify that for certain values there is chaotic and periodic behavior.

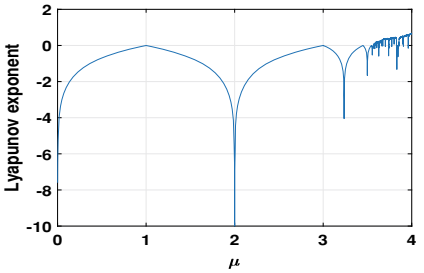
If we take $\mu = 1$ and the initial condition $x_0 = 0.1$, then for $n = 10000$,

$$\lambda(0.1) \approx \frac{1}{n} \sum_{k=0}^{n-1} \ln |x_k(1 - x_k)| = -7.1460 \times 10^{-4}. \tag{8.9}$$

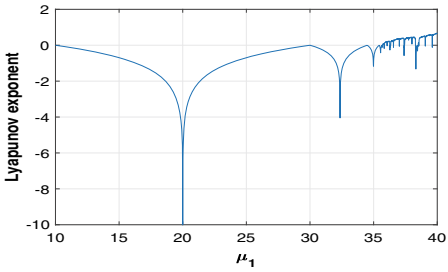
In this case, since $\lambda(0.1) < 0$, there is not chaotic behavior. In the case where $\mu = 4$ and the same initial condition, then for $n = 10000$,

$$\lambda(0.1) \approx 0.6931. \tag{8.10}$$

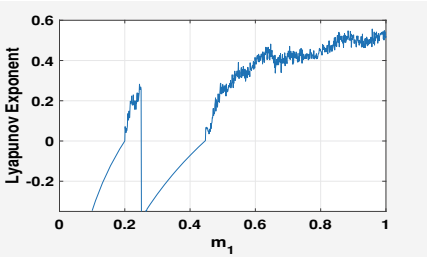
Due to the fact that $\lambda(0.1) > 0$ and that the trajectory is bounded, then at $\mu = 4$, logistic map has chaotic behavior. In general, for any map, these same calculations can be performed for more values of some parameter present in its equation. In particular, in this work, for logistic map, μ was varied from zero to four, taking a step equal to 0.001. The result of these calculations can be seen in Fig. 8.8a. In this figure it is possible observe that, when $\mu > 3.5$, the value of the Lyapunov exponent is greater than zero, leading to the presence of chaotic behavior. It should be noted that not all values of (3.5, 4) have a positive Lyapunov exponent; this is because,



(a) Logistic map.



(b) Vertigo-2 map.



(c) PWL map.

Fig. 8.8 Lyapunov exponents. (**a**) Logistic map. (**b**) Vertigo-2 map. (**c**) PWL map

as seen above, there are regions within this range where interior crisis bifurcations occur. In particular, for $\mu = 4$, there is a positive Lyapunov exponent.

In Fig. 8.8b, we can see the Lyapunov exponent diagram of Vertigo-2 map when μ_1 was varied from ten to forty. It can be appreciated that this is similar to the one of the logistic map, the difference with this one is that for Vertigo-2 map, and the Lyapunov exponents are positive when the value of μ belongs to the range $(37, 40]$, in particular, at $\mu_1 = 40$, the value is positive.

Finally, Fig. 8.8c shows the Lyapunov exponents of PWL map when $m_1 \in [0, 1]$. In this diagram, we can check that the Lyapunov exponent is negative for the two windows given by $0 < m_1 < 0.2$ and $0.25 < m_1 < 0.45$, and it is positive for the intervals: $0.20 < m_1 < 0.25$ and $0.45 < m_1 < 1$. Taking this into account, in these last two intervals, the chaotic behavior is verified because all the orbits are bounded.

Table 8.2 shows the largest exponent for each of the three maps within the range in which the bifurcation parameter was varied. As can be seen, the logistic map and the Vertigo-2 map have a very similar value and it is larger than the Lyapunov exponent of the PWL. Thus, for the first maps, the divergence rate between two close orbits is higher than the one of the PWL map.

Table 8.2 Values of the maximum Lyapunov exponent

Map	Max LE	Value of parameter
Logistic map	0.6931	$\mu = 4$
Vertigo-2 map	0.6933	$\mu_1 = 40$
PWL map	0.5709	$m_1 = 1$

8.2.6 Average Measurement to Sensitivity to Initial Conditions

Chaotic maps have the peculiarity of extreme sensitivity to initial conditions. In the previous section, this was verified by a positive value in the Lyapunov exponent, where a larger value indicated a larger divergence. Although the three maps described have values in the parameters for which chaos is present, the sensitivity to initial conditions is different in each of them. In this part, we will describe another method to measure this particularity.

In general, for a map, the method consists first in selecting 1000 different arbitrary initial conditions x_{0_i}, as well as 1000 closed conditions $x'_{0_i} = x_{0_i} + \epsilon$, where $\epsilon = 1 \times 10^{-14}$ and $i \in \{1, \ldots, 1000\}$. Next, we calculate the distances:

$$d_i^n = |f^n(x_{0_i}) - f^n(x'_{0_i})|, \tag{8.11}$$

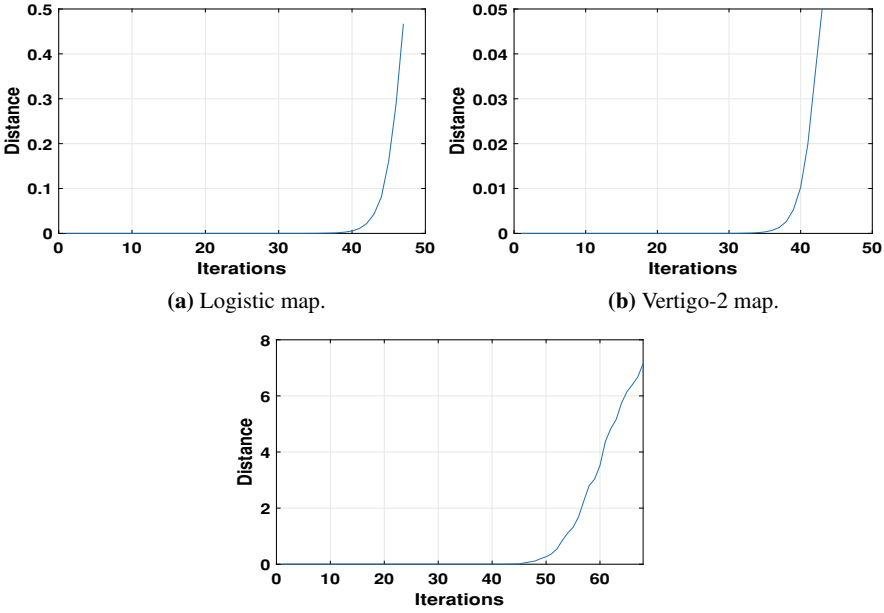
$$\overline{d^n} = \frac{1}{1000} \sum_{i=1}^{1000} d_i^n, \tag{8.12}$$

up to some n in which it is satisfied that $\overline{d^n} \geq d_\alpha$, where the d_α is a threshold given by

$$d_\alpha = \frac{1}{1000} \sum_{k=0}^{1000} |f^k(\overline{x_0}) - f^k(\overline{x'_0})|, \qquad \overline{x_0} = 0.11. \tag{8.13}$$

When this happen, we say that map presents sensitivity to initial conditions at the n iteration.

The method described above was applied to each of the three one-dimensional maps, choosing in every case, the one with the highest value of the Lyapunov exponent, i.e., we took the three maps described in Table 8.2. Figure 8.9 exhibits the graph of the necessary number of iterations to reach the d_α value for each map. As can be seen in Table 8.3, logistic map and Vertigo-2 map need less iterations than PWL map for achieving this target. Then, we can reaffirm that these maps have a higher divergence and a greater sensitivity to initial conditions than the PWL map.



(a) Logistic map. **(b)** Vertigo-2 map.

(c) PWL map.

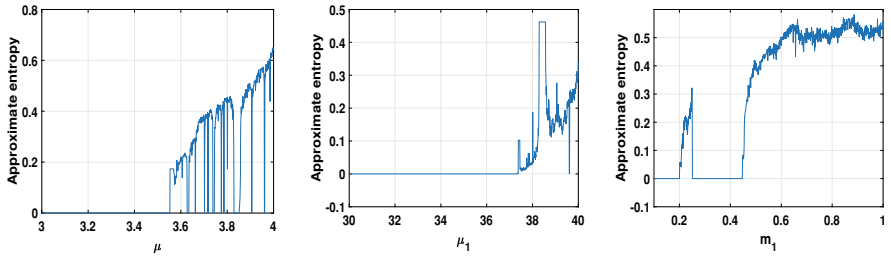
Fig. 8.9 Sensitivity to initial conditions. (**a**) Logistic map. (**b**) Vertigo-2 map. (**c**) PWL map

Table 8.3 Number of iterations necessary to manifest sensitivity to initial conditions

Map	$\overline{x_0}$	d_α	Iterations
Logistic map	0.11	0.47	46
Vertigo-2 map	0.11	0.050	44
PWL map	0.11	7.10	68

8.2.7 Entropy Analysis

Entropy is another way to measure how much disorder and complexity are present in the maps. One effective way to calculate the entropy for chaotic series is the approximate entropy, which indicates that the higher the entropy value, the more chaotic and complex is the dynamical system. In Fig. 8.10, we show the calculation of the approximate entropy for the three maps and how it changes due to the variation of its control parameter. From here, we can remark that the entropy value for PWL map is more regular when varying the parameter m_1 than the entropy values of the remaining pair of maps, mainly the logistic one, which has values alternating between positive entropy and entropy equal to zero. In Table 8.4, we can observe that logistic map has the highest approximate entropy of the three maps. Of the two maps without fixed points, PWL has a higher value, and hence its dynamics is more complex than that of Vertigo-2 map, but slightly lower than that of the logistic map.



(a) Logistic map. **(b)** Vertigo-2 map. **(c)** PWL map.

Fig. 8.10 Approximate entropy. (**a**) Logistic map. (**b**) Vertigo-2 map. (**c**) PWL map

Table 8.4 Maximum approximate entropy of each of maps

Map	Max ApEn	Value of parameter
Logistic map	0.650	$\mu = 4.00$
Vertigo-2 map	0.460	$\mu_1 = 38.3$
PWL map	0.590	$m_1 = 0.89$

8.3 Bidimensional Maps

8.3.1 The Hénon Map

The first map presented here is the Hénon map [28] and is defined by the equation

$$\begin{aligned} x_{n+1} &= 1 - ax_n^2 + y_n, \\ y_{n+1} &= bx_n. \end{aligned} \tag{8.14}$$

If we set the following parameters: $a = 1.4$, $b = 0.3$, $x_0 = 0$, and $y_0 = 0$, then the system shows chaotic behavior. For these values, Hénon proved the generation of what is known as a strange attractor. The plot of this strange attractor is showed in Fig. 8.11.

Fixed Points The Hénon map has two fixed points $x^*_{1,2}$

$$x^*_{1,2} = \frac{-(1-b) \pm \sqrt{(1-b)^2 + 4a)}}{2a}. \tag{8.15}$$

Map Inspirited by Hénon Map (IHM Map)

The second bidimensional map presented here and as mentioned in [16] is a map inspired in Hénon's map. In that work, several cases are dealt with, which depend on the existence or not of fixed points. In general, the equation that describes its dynamics is

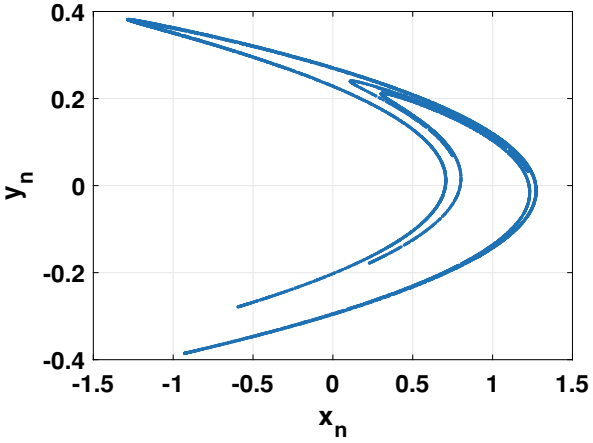


Fig. 8.11 Attractor of Hénon map

$$\begin{aligned} x_{n+1} &= y_n, \\ y_{n+1} &= a_1 x_n + a_2 y_n + a_3 x_n^2 + a_4 y_n^2 + a_5 x_n y_n + a_6. \end{aligned} \tag{8.16}$$

Fixed Points In this chapter, we will focus on the case where there are no fixed points. Equating

$$\begin{aligned} x_n &= y_n, \\ y_n &= a_1 x_n + a_2 y_n + a_3 x_n^2 + a_4 y_n^2 + a_5 x_n y_n + a_6. \end{aligned} \tag{8.17}$$

Then, under the following considerations, (8.17) has no solution, and therefore (8.16) has no fixed points:

- If $a_3 + a_4 + a_5 = 0$, $a_1 + a_2 - 1 = 0$, and $a_6 \neq 0$.
- If $a_3 + a_4 + a_5 \neq 0$ and $\bigtriangleup < 0$.

As previously mentioned, if the basin of attraction of any attractor does not intersect with any neighborhood of the fixed point, then we have a hidden attractor. Since for this case there are no fixed points, the attractors generated will be of this type. Figure 8.12 displays the hidden attractor generated from (8.16), when $a_1 = 1, a_2 = 0, a_3 = -0.6, a_4 = 0.74, a_5 = -0.14$, and $a_6 = -0.33$.

$$\begin{aligned} x_{n+1} &= y_n, \\ y_{n+1} &= x_n - 0.6 x_n^2 + 0.74 y_n^2 - 0.14 x_n y_n - 0.33. \end{aligned} \tag{8.18}$$

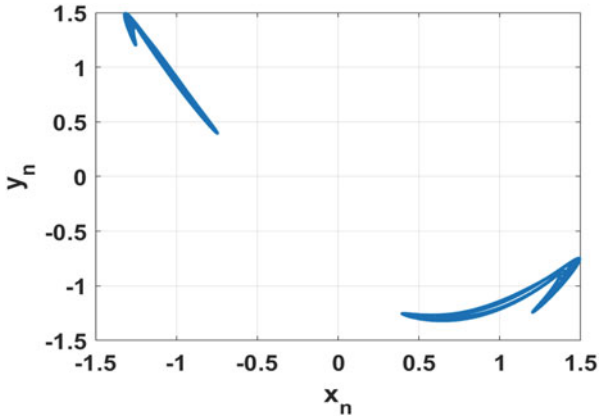


Fig. 8.12 Hidden attractor of IHM map

Quadratic Map

The third map discussed here is one of the cases presented in [29]. In this chapter, we will refer to this map as the quadratic map, and its iterated equation is

$$\begin{aligned} x_{n+1} &= y_n + x_n, \\ y_{n+1} &= y_n[a_1 x_n + a_2 y_n + a_3] + y_n + a_4 x_n^2 + a_5 x_n + a_6. \end{aligned} \tag{8.19}$$

Fixed Points Under the condition $a_5^2 + 4a_4a_6 < 0$, (8.19) represents a quadratic map with no fixed points. In Fig. 8.13, we can see the hidden attractor generated when the parameters for (8.19) are $a_1 = -1, a_2 = 2, a_3 = 0, a_4 = 0.1, a_5 = 0$, and $a_6 = 0.1$.

Map with No Fixed Points (NFP Map)

The last map exposed here is the one proposed in [21]. Its mathematical description is

$$\begin{aligned} x_{n+1} &= y_n + x_n, \\ y_{n+1} &= -y_n x_n + y_n - a|y_n| + bx_n^2 - cy_n^2 + d. \end{aligned} \tag{8.20}$$

Fixed Points The existence of fixed points could be determined by the solution of

$$\begin{aligned} x &= y + x, \\ y &= -yx + y - a|y| + bx^2 - cy^2 + d. \end{aligned} \tag{8.21}$$

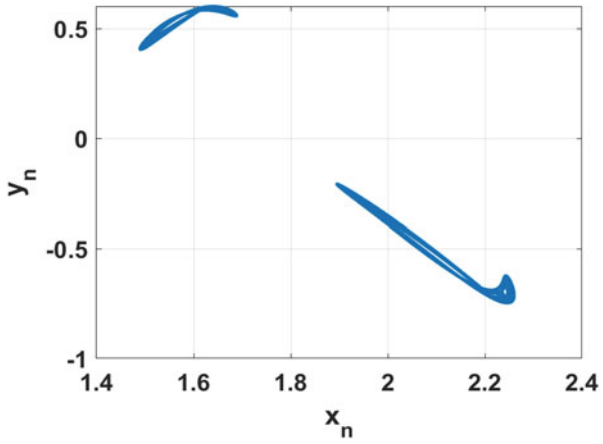


Fig. 8.13 Hidden attractor of the quadratic map

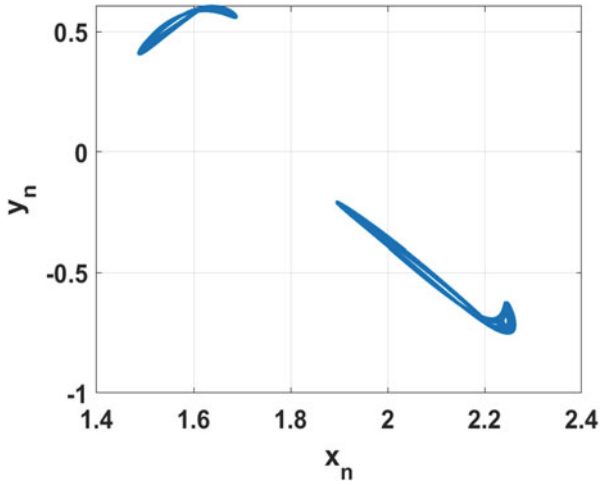
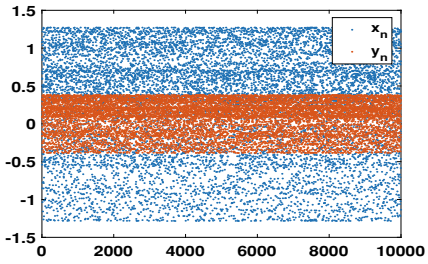


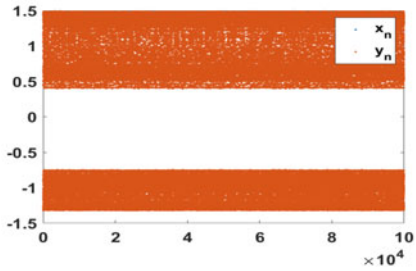
Fig. 8.14 Hidden attractor of the NFP map

From (8.21), y must be zero, and hence $bx^2 + d = 0$. Therefore, in particular, if the values b and d are both positive and $bx^2 + d \neq 0$, then (8.20) has no fixed points.

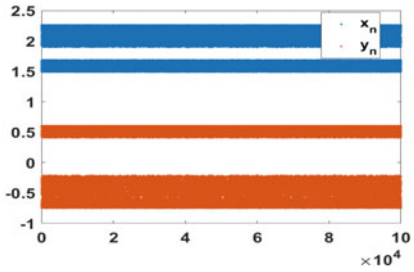
Figure 8.14 shows the hidden attractor for (8.20) when $a = 0.01$, $b = 0.1$, $c = 2$, and $d = 0.1$.



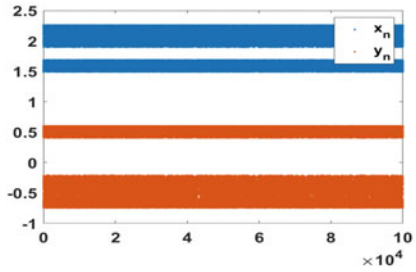
(a) Hénon map.



(b) IHM map.



(c) Quadratic map.



(d) NFP map.

Fig. 8.15 Time series of the four maps. (**a**) Hénon map. (**b**) IHM map. (**c**) Quadratic map. (**d**) NFP map

8.3.2 Time Series

Similar to the previous one-dimensional maps, where we could observe by means of the time series a non-periodic behavior, in Fig. 8.15, we can appreciate a non-periodic trajectory in x_n, as well as in y_n in the four two-dimensional maps. It is also important to note that these trajectories are bounded between the regions where the attractor is present and that they also traverse most of the latter.

8.3.3 Bifurcation Diagrams

In this section, we will perform an analysis of the dynamics for each of the four maps described above. In the same way as in the one-dimensional maps, one parameter will be varied, while the rest will remain fixed, so by realization of bifurcation diagrams, it will be verified for which ones there is chaotic behavior.

In Fig. 8.16a, for the Hénon map, we can observe the bifurcation diagram in x_n when the parameter a varies from 0.1 to 1.4, with an initial condition $x_0 = 0$, $y_0 = 0$. From this, we can detect a route period-doubling to chaos. This same type

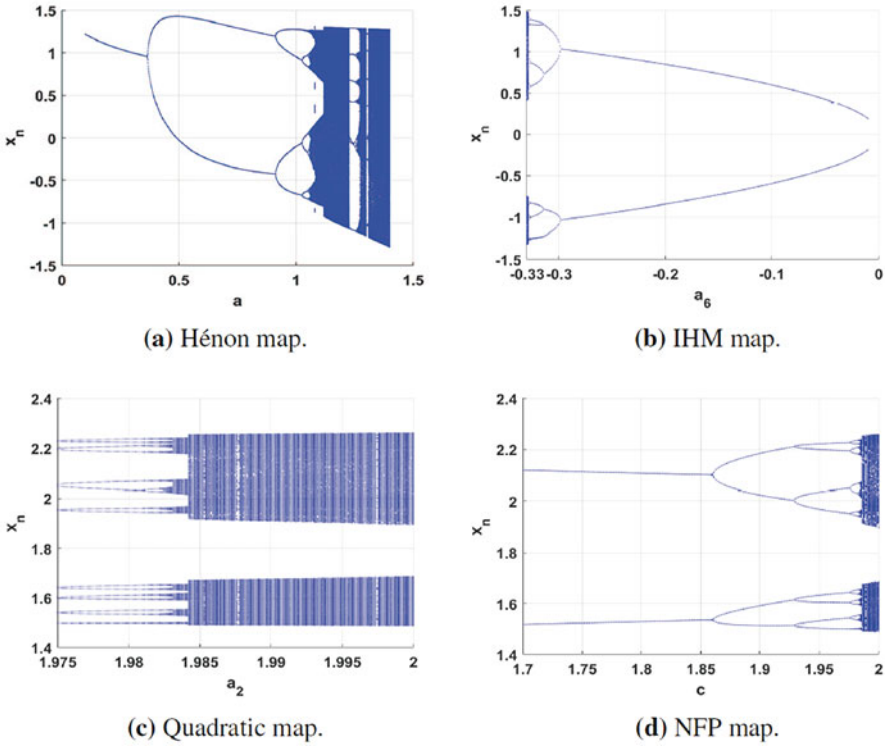


Fig. 8.16 Bifurcation diagrams. (**a**) Hénon map. (**b**) IHM map. (**c**) Quadratic map. (**d**) NFP map

of bifurcation is present in the rest of the three maps. For IHM map, Fig. 8.16b shows the bifurcation diagram, obtained by varying the parameter a_6 between -0.33 and -0.05 and an initial condition $x_0 = -0.78,\ y_0 = 0.45$. Figure 8.16c depicts the bifurcation diagram of quadratic map when the parameter a_2 is varied between 1.975 and 2, a range in which there are no fixed points; the initial value is $x_0 = 1.5,\ y_0 = 0.55$. Finally, Fig. 8.18d shows the bifurcation diagram for NFP map, generated by increasing the parameter c from 1.7 until 2, and the initial condition was $x_0 = 1.5,\ y_0 = 0.5$.

Table 8.5 shows the intervals in which each of these behaviors is manifested. From here, it is possible to differentiate that the Hénon map is the one with the highest range where chaotic dynamics is developed; in addition, the quadratic map and NFP map have the same interval of chaotic behavior, and IHM map has the lowest of the four.

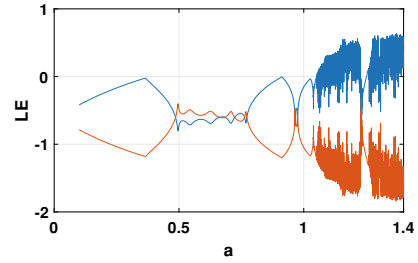
Table 8.5 Chaotic and periodic behavior for the maps

Map	First period	Bifurcation	Periodic behavior	Chaos
Hénon	1	Periodic-doubling interior crisis	(0, 1.1)	(1.1, 1.4]
IHM	2	Periodic-doubling interior crisis	$(-0.326, 0)$	$[-0.33, -0.326)$
NFP	2	Periodic-doubling interior crisis	(0, 1.985)	(1.985, 2]
Quadratic map	2	Periodic-doubling interior crisis	(0, 1.985)	(1.985, 2]

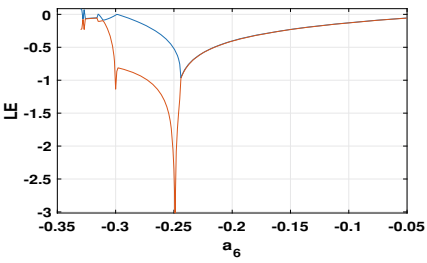
8.3.4 Lyapunov Exponents Spectrum

In this section, we are going to show the existence of chaotic behavior through the calculation of the Lyapunov exponents for the four bidimensional maps. The method that we used to obtain the spectrum of Lyapunov exponents was the discrete QR algorithm. The results of these estimates can be seen in Fig. 8.17. Firstly, and as previously seen, for the Hénon map, there are regions with non-periodic behavior. In Fig. 8.17a, starting at about 1.05, a positive Lyapunov exponent is observed with a tendency to increase in value as the parameter a is modified and reaching a maximum value at $a = 4$; then, for these values of a, chaotic dynamics is present. In Fig. 8.17b, we can appreciate the Lyapunov exponent diagram of IHM map. The existence of chaotic dynamics is evidenced when the value of a_2 belongs to interval $[-0.33, -0.326)$. Figure 8.17c shows the Lyapunov exponents of quadratic map; one of them is positive in the interval $(1.978, 2)$. Furthermore, since the orbits in this interval are bounded, then, there is chaotic behavior. For NFP map, and taking c as the variation parameter, it can be seen in Fig. 8.17d that as in the previous map, one of the exponents is positive and the other negative for the same values for both a and c. Thus, chaos also exists.

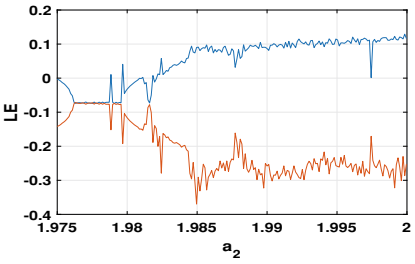
If we compare the values at which the maps show chaos, we see that these are consistent with those obtained in the bifurcation diagrams and for whom non-periodic dynamics are detected. In Table 8.6 is given the largest positive value of the Lyapunov exponent and its corresponding negative value obtained for each of the four maps. In such a table, it stands out that the Hénon map is the one with the largest exponent of the four and that quadratic map and NFP map have very similar values, just when their parameters a and c, respectively, are two. From this, we can say that the divergence for the Hénon map is the highest of the four and the IHM has the lowest.



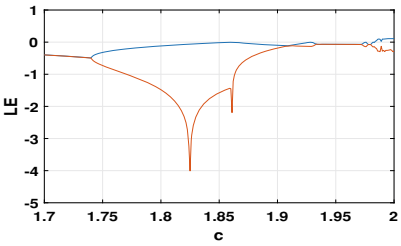
(a) Hénon map.



(b) IHM map.



(c) Quadratic map.



(d) NFP map.

Fig. 8.17 Lyapunov exponents spectrum. (**a**) Hénon map. (**b**) IHM map. (**c**) Quadratic map. (**d**) NFP map

Table 8.6 Values of the maximum Lyapunov exponent

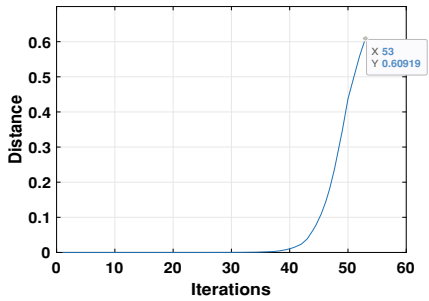
Map	Max LE x_n	Corresponding LE y_n	Value of parameter
Hénon map	0.4270	-1.6310	$a = 4$
IHM	0.08629	-0.2304	$a_6 = -0.33$
Quadratic map	0.1291	-0.2526	$a_2 = 2$
NFP map	0.1150	-0.2760	$c = 2$

8.3.5 Average Measurement to Sensitivity to Initial Conditions

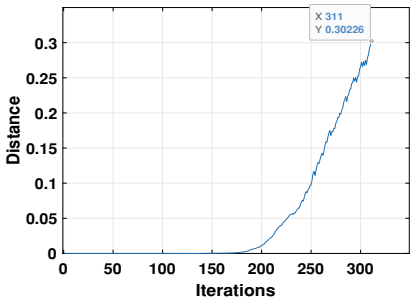
In this section, we will analyze the sensitivity to initial conditions with the same method as in Sect. 8.2.4. Since here we deal with two-dimensional maps, the distance used will be

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}. \tag{8.22}$$

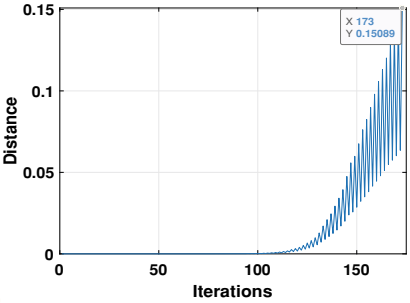
Again, 1000 arbitrary initial conditions x_0 and y_0 and their respective close initial conditions x_{0_ϵ} and y_{0_ϵ} were taken, where $\epsilon = 1 \times 10^{-14}$. Figure 8.18 exhibits the graph of the necessary number of iterations to reach the d_α value for each map. As we can check in Table 8.7, as well as in Fig. 8.18a, the Hénon map is the one that



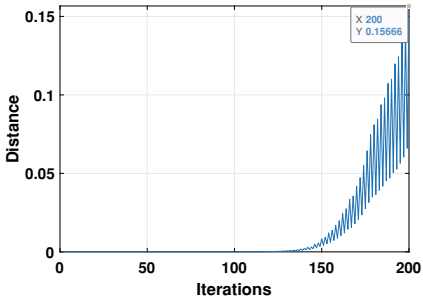
(a) Hénon map.



(b) IHM map.



(c) Quadratic map.



(d) NFP map.

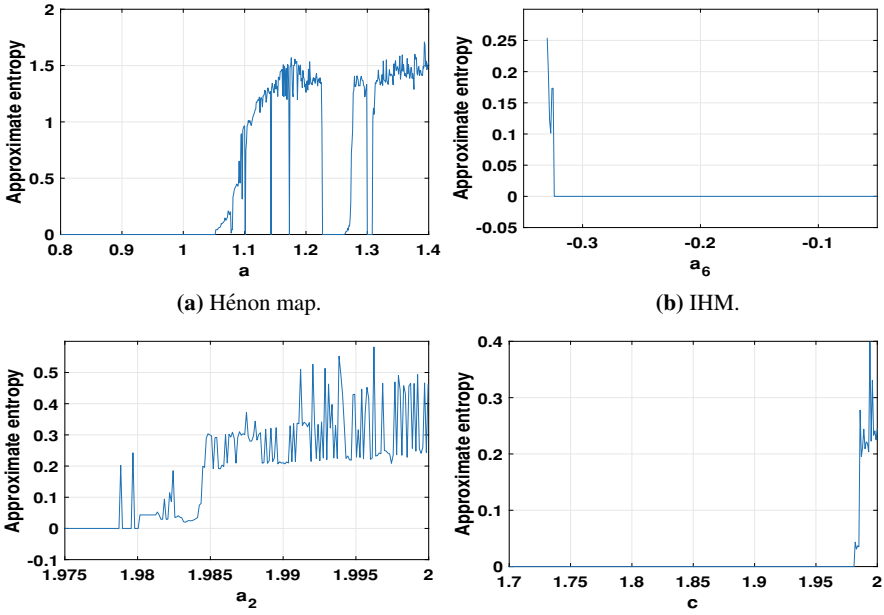
Fig. 8.18 Graph of number of iterations vs distance. (**a**) Hénon map. (**b**) IHM map. (**c**) Quadratic map. (**d**) NFP map

Table 8.7 Iterations necessary to manifest sensitivity to IC

Map	d_α	Iterations
Hénon map	0.6092	53
IHM	0.3023	311
Quadratic map	0.1509	173
NFP map	0.1567	200

achieves in less iterations to overcome its d_α value. Quadratic and NFP maps need more but approximately the same iterations to reach its threshold.

Table 8.7 and Fig. 8.18c and d give evidence of this fact. The worst performance shown in this table, as well as in Fig. 8.18b, is for IHM map. These results agree with the previous ones, where the order from the highest to the lowest positive Lyapunov exponent was the Hénon map, quadratic map and NFP map, and finally IHM map.



(a) Hénon map. **(b)** IHM.

(c) Quadratic map. **(d)** NFP map.

Fig. 8.19 Approximate entropy. (**a**) Hénon map. (**b**) IHM. (**c**) Quadratic map. (**d**) NFP map

Table 8.8 Maximum approximate entropy of each of maps.

Map	Max ApEn	Value of parameter
Hénon map	1.650	$a = 4.00$
IHM map	0.2543	$a_6 = -0.33$
Quadratic map	0.4773	$a_2 = 2.00$
NFP map	0.3990	$c = 1.994$

8.3.6 Entropy of Bidimensional Maps

In this section, the calculation of the approximate entropy will allow us to re-evaluate the divergence and the level of complexity and disorder for the four two-dimensional maps. In Fig. 8.19, we show the variation of the approximate entropy for the four maps. Again, the Hénon map is found to have a higher range in its parameter with respect to the rest of the maps in which there is positive entropy. In addition, in Fig. 8.19a–d, it is verified that these ranges coincide with those obtained in the bifurcation diagrams as well as Lyapunov exponents where there was evidence of chaotic dynamics. Finally, in Table 8.8, the highest entropy values within the ranges in which each of the parameters is varied are displayed.

8.4 Conclusions

In this work, we have performed a comparative analysis of some chaotic features to unidimensional and bidimensional maps, both with fixed points and without fixed points. For the case of one-dimensional maps, the following have been used: cobweb plots, bifurcation diagrams, Lyapunov exponents, histograms, time series, and entropy calculation. While for the analysis of the two-dimensional maps, the following have been implemented: phase space diagrams, time series, bifurcation diagrams, Lyapunov exponents, and entropy calculation. Finally, comparative tables were established in which the differences and similarities between the studied maps could be contrasted.

For the one-dimensional maps, the results show that the chaotic properties between the logistic map and the Vertigo-2 map are very similar; this is mainly due to the fact that the Vertigo-2 map is a scaling of the logistic map but does not have fixed points. In contrast, between the logistic map and the PWL map without fixed points, we have found differences, mainly in the route to chaos, where this last one does not show period-doubling bifurcation; in addition, between these two maps, the divergence, the sensitivity to initial conditions, and the degree of complexity and disorder are slightly higher for the logistic map. However, from the results, we can see that the PWL map without fixed points has a larger range in its parameter in which chaotic behavior is presented with respect to the logistic map and the Vertigo-2 map.

In the case of two-dimensional maps, it could be observed that all four maps follow a route to chaos through a period-doubling bifurcation. However, there were noticeable differences between the Hénon map and the three maps without fixed points. The highest difference was found between the Hénon map and the Hénon-inspired map. These differences can be appreciated in the divergence and sensitivity to initial conditions, the transitivity of the orbits, and in the range where chaos occurs. In other words, the Hénon map has a higher complexity in its dynamics than the other three maps.

In general, in the case of both one- and two-dimensional maps, the comparative results may contribute to the choice of the most appropriate parameters and maps in future applications, such as in the generation of pseudorandom sequences or in encryption schemes.

Data Availability
The data used to support the findings of this study are included within the chapter.

Conflicts of Interest
The authors declare that there are no conflicts of interest regarding the publication of this work.

Acknowledgments C. García-Grimaldo is thankful to CONACYT for the scholarships granted. Eric Campos acknowledges CONACYT for the financial support through project no. A1-S-30433.

References

1. G. Chen, X. Yu, *Chaos Control: Theory and Applications* (Springer, Berlin, 2003)
2. L. Chua, M. Komuro, T. Matsumoto, The double scroll family. IEEE Trans. Circuits Syst. **33**, 1072–1118 (1986)
3. G.M. Mindlin, R. Gilmore, Topological analysis and synthesis of chaotic time series. Physica D Nonlinear Phenomena **58**, 229–242 (1992)
4. P. Grassberger, I. Procaccia, Estimation of the Kolmogorov entropy from a chaotic signal. Phys. Rev. A, **28**, 2591–2593 (1983)
5. T. Zhou, Y. Tang, G. Chen, Complex dynamical behaviors of the chaotic Chen's system. Int. J. Bifurc. Chaos **13**, 2561–2574 (2003)
6. R. Bowen, D. Ruelle, The ergodic theory of AxiomA flows. Invent Math. **29**, 181–202 (1975)
7. M. García-Martínez, E. Campos-Cantón, Pseudo-random bit generator based on multi-modal maps. Nonlinear Dyn. **82**, 2119–2131 (2015)
8. M. García-Martínez, L.J. Ontañón-García, E. Campos-Cantón, S. Celikovský, Hyperchaotic encryption based on multiscroll piecewise linear systems. Appl. Math. Comput. **270**, 413–424 (2015)
9. E. Lorenz, Deterministic nonperiodic flow. J. Atmos. Sci. **20**, 130–141 (1963)
10. L.P. Shil'nikov, A case of the existence of a countable number of periodic motions. Soviet Math. Dokl. **6**, 163–166 (1965)
11. S. Jafari, J.C. Sprott, S.M.R.H. Golpayegani, Elementary quadratic chaotic flows with no equilibria. Physics Lett. A **3779**, 699–702 (2013)
12. R.J. Escalante-González, E. Campos-Cantón, Generation of chaotic attractors without equilibria via piecewise linear systems. Int. J. Mod. Phys. C **28**, 1750008 (2017)
13. R.J. Escalante-González, E. Campos-Cantón, A class of piecewise linear systems without equilibria with 3-D grid multiscroll chaotic attractors. IEEE Trans. Circuits Syst. II-Express Briefs **66**, 1456–1460 (2019)
14. S. Jafari, V.-T. Pham, S.M.R.H. Golpayegani, M. Moghtadaei, S.T. Kingni, The relationship between chaotic maps and some chaotic systems with hidden attractors. Int. J. Bifurc. Chaos **26**, 1650211–1–8 (2016)
15. D. Lambić, A new discrete chaotic map based on the composition of permutations. Chaos Solitons Fractals **78**, 245–248 (2015)
16. H. Jiang, L. Liu, Z. Wei, L. Zhang, Hidden chaotic attractors in a class of two-dimensional maps. Nonlinear Dyn. **85**, 2719–2727 (2016)
17. C. Wang, Q. Ding, A new two-dimensional map with hidden attractors. Entropy **20**, 322 (2018)
18. C. García-Grimaldo, E. Campos-Cantón, Chaotic features of a class of discrete maps without fixed points. Int. J. Bifurc. Chaos **31**, 2150200 (2021)
19. D. Dudkowski, S. Jafari, T. Kapitaniak, N.V. Kuznetsov, G.A. Leonov, A. Prasad, Hidden attractors in dynamical systems. Phys. Rep. **637**, 1–50 (2016)
20. S. Jafari, V.-T. Pham, T. Kapitaniak, Multiscroll chaotic sea obtained from a simple 3D system without equilibrium. Int. J. Bifurc. Chaos **26**, 1650031–1–7 (2016)
21. V.V. Huynh, A. Ouannas, X. Wang, V.-T. Pham, X.Q. Nguyen, F.E. Alsaadi, Chaotic map with no fixed points: entropy, implementation and control. Entropy **21**, 279 (2019)
22. C. Li, D. Wang, An attractor with invariable Lyapunov exponent spectrum and its Jerk circuit implementation. Acta Phys. Sin. (Chinese Edition) **58**, 764–770 (2009)
23. C.B. Li, S. Chen, H.Q. Zhu, Circuit implementation and synchronization of an improved system with invariable Lyapunov exponent spectrum. Acta Phys. Sin. (Chinese Edition) **58**, 2255–2265 (2009)
24. C.B. Li, H.K. Wang, An extension system with constant Lyapunov exponent spectrum and its evolvement study. Acta Phys. Sin. (Chinese Edition) **58**, 7514–7524 (2015)
25. C. Li, J.C. Sprott, A. Akgul, H.H.C. Lu, A new chaotic oscillator with free control. Chaos **27**, 083101 (2017)

26. R. May, Simple mathematical models with very complicated dynamics. Nature **261**, 459–467 (1976)
27. K.T. Alligood, T.D. Sauer, J.A. Yorke, *Chaos: An Introduction to Dynamical Systems* (Springer, New York, 2000)
28. M. Hénon, A two-dimensional mapping with a strange attractor. Commun. Math. Phys. **50**, 69–77 (1976)
29. S. Panahi, J. Sprott, S. Jafari, Two simplest quadratic chaotic maps without equilibrium. Int. J. Bifurc. Chaos **28**, 1850144 (2018)

Chapter 9
A New 4-D Hyperchaotic System with No Balance Point, Its Bifurcation Analysis, Multi-Stability, Circuit Simulation, and FPGA Realization

Sundarapandian Vaidyanathan, Esteban Tlelo-Cuautle, Omar Guillén-Fernández, Khaled Benkouider, and Aceng Sambas

Abstract This work proposes a new 4-D hyperchaotic system with three quadratic nonlinear terms. We establish that the proposed system has no balance point. We deduce that the system has hidden attractors. We carry out a dynamic analysis of the new system with bifurcation diagrams and Lyapunov exponents. We show that the new system has multi-stability and coexisting attractors. Using MultiSim Version 14, we design an electronic circuit of the new hyperchaotic system. The new 4-D hyperchaotic system with no balance point is verified under an implementation using a field-programmable gate array (FPGA). We show the block diagrams' descriptions of the system by applying three one-step numerical methods, which consist of multipliers, adders, and subtractors, and we list the hardware resources required for each numerical method. Finally, we show the experimental hyperchaotic attractors, which are in good agreement with simulation results.

Keywords Bifurcation · Circuit design · Hyperchaotic system · Hyperchaos · Multi-stability · FPGA realization

S. Vaidyanathan (✉)
Vel Tech University, Chennai, Tamil Nadu, India
e-mail: sundar@veltech.edu.in

E. Tlelo-Cuautle · O. Guillén-Fernández
INAOE, Tonantzintla, Puebla, Mexico
e-mail: etlelo@inaoep.mx

K. Benkouider
University of Jijel, Jijel, Algeria

A. Sambas
Universitas Muhammadiyah Tasikmalaya, Tasikmalaya, Indonesia
e-mail: acengs@umtas.ac.id

G. Huerta Cuellar et al. (eds.), *Complex Systems and Their Applications*,
https://doi.org/10.1007/978-3-031-02472-6_9

9.1 Introduction

Due to their high complexity and randomness, hyperchaotic dynamical systems find several applications in engineering, such as: memristors [1–3], lasers [4, 5], robotics [6], cryptosystems [7, 8], etc. Chaotic and hyperchaotic oscillators have been a hot topic for research during the last years due to their usefulness in the development of chaotic secure communication systems and other applications that have been implemented using either analog or digital electronics, as already shown in [9, 10].

Hyperchaotic attractors can be broadly divided into two categories, viz. self-excited and hidden hyperchaotic attractors. Hyperchaotic attractors generated by an unstable balance point belong to the class of *self-excited* hyperchaotic attractors [11]. Hyperchaotic attractors with no balance point [12], or with a stable balance point [13], or possessing infinitely many balance points [14], belong to the class of *hidden* hyperchaotic attractors [11].

In this chapter, we propose a new 4-D hyperchaotic system having three quadratic nonlinear terms. We shall exhibit that the proposed hyperchaotic system does not have any balance point for all non-zero values of the system parameters. Hence, we deduce that the proposed system has hidden hyperchaotic attractors.

We carry out an extensive bifurcation analysis of the new hyperchaotic system with no balance point. Bifurcation analysis is a miscellaneous technique to investigate the dynamic behavior of nonlinear systems with respect to variation of the system parameters [15, 16].

Multi-stability is basically the phenomenon of a chaotic or hyperchaotic system of coexisting attractors for the same values of the parameters but different values of initial states [17–19]. In this chapter, we demonstrate that the new hyperchaotic system has the special property of multi-stability with coexisting attractors.

Circuit implementations of hyperchaotic systems enable their real-world implementation for various engineering applications [20–22]. In Sect. 9.2, we show that by using MultiSim version 14, one can design an electronic circuit for the proposed hyperchaotic system with no balance point. Section 9.3 provides details on generating the bifurcation diagrams and Lyapunov exponents' spectrums of the new 4-D system in order to investigate the dynamical behavior. In addition, this section demonstrates the complex phenomenon of coexistence of attractors. Section 9.4 provides guidelines to design the electronic circuit using amplifiers. The implementation of the new 4-D hyperchaotic system with no balance, using a field-programmable gate array (FPGA), is given in Sect. 9.5, where experimental observations of the attractors are shown using a Teledyne LeCroy oscilloscope. Finally, the conclusions are summarized in Sect. 9.6.

9.2 A New 4-D Hyperchaotic System with No Balance Point

In this section, we introduce the mathematical model of a new 4-D system with the dynamical equations given as below:

$$\begin{cases} \dot{y}_1 = -a(y_1 + y_2) + y_2 y_3 + c y_4 \\ \dot{y}_2 = -y_1 - a y_1 y_3 - c y_4 \\ \dot{y}_3 = b + y_1 y_2 \\ \dot{y}_4 = y_2 \end{cases} \tag{9.1}$$

In (9.1), the coefficients a, b, c are system constants and we suppose that $b \neq 0$. We use Y to denote the states of the dynamical system (9.1), in which four state variables exist, i.e., $Y = (y_1, y_2, y_3, y_4)$.

It shall be established in this section that the proposed 4-D hyperchaotic system exhibits a hidden hyperchaotic attractor for the parameter values: $a = 15$, $b = 20$, and $c = 2$.

The solution of the dynamical system by performing numerical simulations is obtained choosing the initial states of the 4-D system (9.1) as:

$$y_1(0) = 0.3,\ y_2(0) = 0.1,\ y_3(0) = 0.3,\ y_4(0) = 0.1 \tag{9.2}$$

The Lyapunov characteristic exponents (LCEs) of the new 4-D system (9.1) for the parameter vector $(a, b, c) = (15, 20, 2)$ and initial state $Y(0) = (0.3, 0.1, 0.3, 0.1)$ were numerically estimated in MatLab for $T = 1E5$ seconds as:

$$\mu_1 = 3.46788,\ \mu_2 = 0.14108,\ \mu_3 = 0,\ \mu_4 = -18.57677 \tag{9.3}$$

As one can see, since there are two positive LCE values in (9.3), and the total of LCE values in (9.1) is negative, then we conclude that the new 4-D system (9.1) has a hyperchaotic attractor with dissipativity.

As a next step, we calculate the equilibrium or balance points for the new 4-D hyperchaotic system, by solving the following system of equations:

$$-a(y_1 + y_2) + y_2 y_3 + c y_4 = 0 \tag{9.4a}$$

$$-y_1 - a y_1 y_3 - c y_4 = 0 \tag{9.4b}$$

$$b + y_1 y_2 = 0 \tag{9.4c}$$

$$y_2 = 0 \tag{9.4d}$$

Using (9.4d), we deduce that $y_2 = 0$. Substituting $y_2 = 0$ into (9.4c), we deduce that $b = 0$, which contradicts the assumption that $b \neq 0$. Therefore, the new 4-D hyperchaotic system (9.1) has no balance point for $b \neq 0$. As shown that the hyperchaotic behavior arises when $(a, b, c) = (15, 20, 2)$.

As the new 4-D hyperchaotic system (9.1) has no balance point, it follows that it has hidden attractors.

The Matlab plots of the new hyperchaotic system (9.1) for the hyperchaotic case $a = 15$, $b = 20$, $c = 2$, and $Y(0) = (0.3, 0.1, 0.3, 0.1)$ are shown in Fig. 9.1.

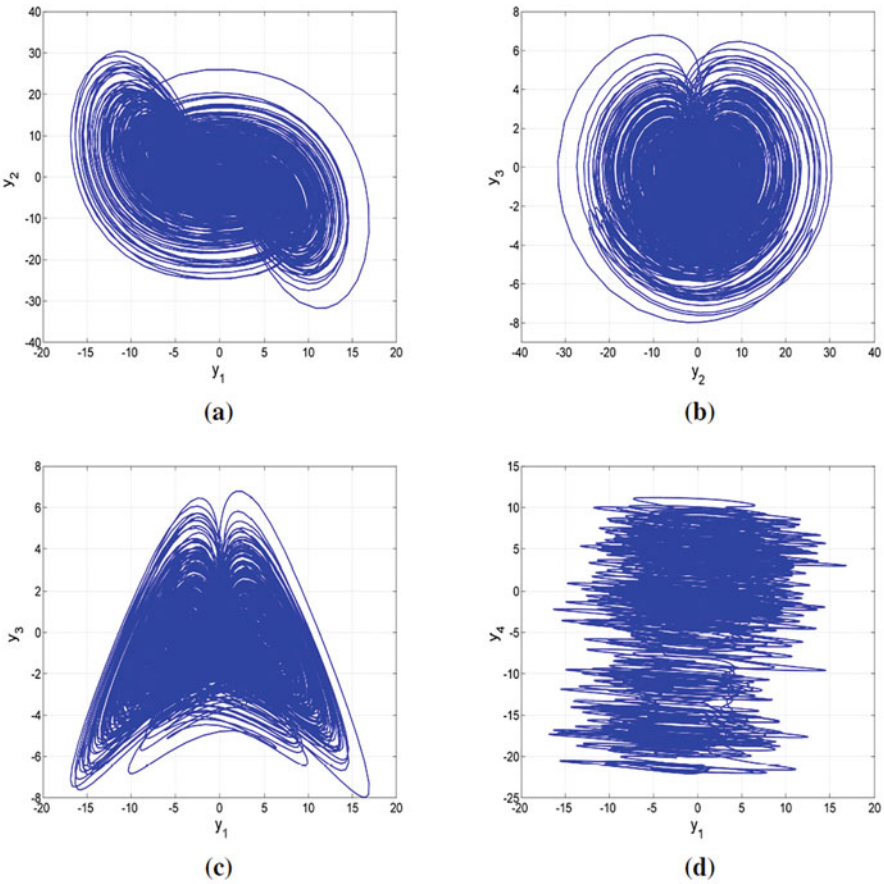


Fig. 9.1 Matlab phase-space portraits of the new 4-D hyperchaotic system (9.1) with no balance point for $a = 15$, $b = 20$, $c = 2$, and $Y(0) = (0.3, 0.1, 0.3, 0.1)$. (**a**) (y_1, y_2)-plane. (**b**) (y_2, y_3)-plane. (**c**) (y_1, y_3)-plane. (**d**) (y_1, y_4)-plane

9.3 Bifurcation Analysis of the New 4-D Hyperchaotic System with No Balance Point

In this section, the bifurcation diagrams and Lyapunov exponents' spectrums are calculated and plotted in order to investigate the dynamical behavior of the new 4-D hyperchaotic system (9.1). This helps to observe the switching of its dynamics between periodic, quasi-periodic, chaos, and hyperchaos when changing the values of the system parameters. For instance, we show the dynamics when each parameter a, b, c is swept.

9.3.1 Bifurcation Analysis When a Varies

The new 4-D hyperchaotic system (9.1) has three parameters a, b, and c. By setting $b = 20$ and $c = 2$, and by varying a in the interval [15, 100], one can observe different dynamical behaviors. Because a takes different values, the Lyapunov exponent values will also vary.

The corresponding bifurcation diagram and Lyapunov exponents' spectrum are obtained numerically by increasing the parameter a from 15 to 100, with $Y(0) = (0.3, 0.1, 0.3, 0.1)$ and then the results are plotted in Fig. 9.2a and b, respectively.

It is observed from Fig. 9.2 that the new 4-D hyperchaotic system (9.1) can exhibit hyperchaos for a wide range of values of parameter a. It can also exhibit chaos, quasi-periodic, and periodic behavior for some parameter intervals.

When $a \in [15, 50]$, the new system (9.1) displays hidden hyperchaotic behavior with a very high value of maximal Lyapunov exponents (MLE greater than 4) indicating high complexity of the hyperchaotic dynamics.

Let us take parameter $a = 40$ and keep $b = 20$ and $c = 2$. Then, the hyperchaotic attractor of (9.1) in (y_1, y_2) plane is plotted in Fig. 9.3a and its corresponding Lyapunov exponents are obtained as follows:

$$\mu_1 = 4.574,\ \mu_2 = 0.046,\ \mu_3 = 0,\ \mu_4 = -44.613 \tag{9.5}$$

In this case, the Lyapunov dimension of (9.1) is derived as

$$D_L = 3 + \frac{\mu_1 + \mu_2 + \mu_3}{|\mu_4|} = 3.0136 \tag{9.6}$$

When a varies in the range [50, 70], and keeping $b = 20$ and $c = 2$, the new 4-D hyperchaotic system (9.1) has only one positive Lyapunov exponent, which means that it presents hidden chaotic attractor with a high value of maximal Lyapunov exponent ($MLE > 3$) indicating high complexity of system behavior.

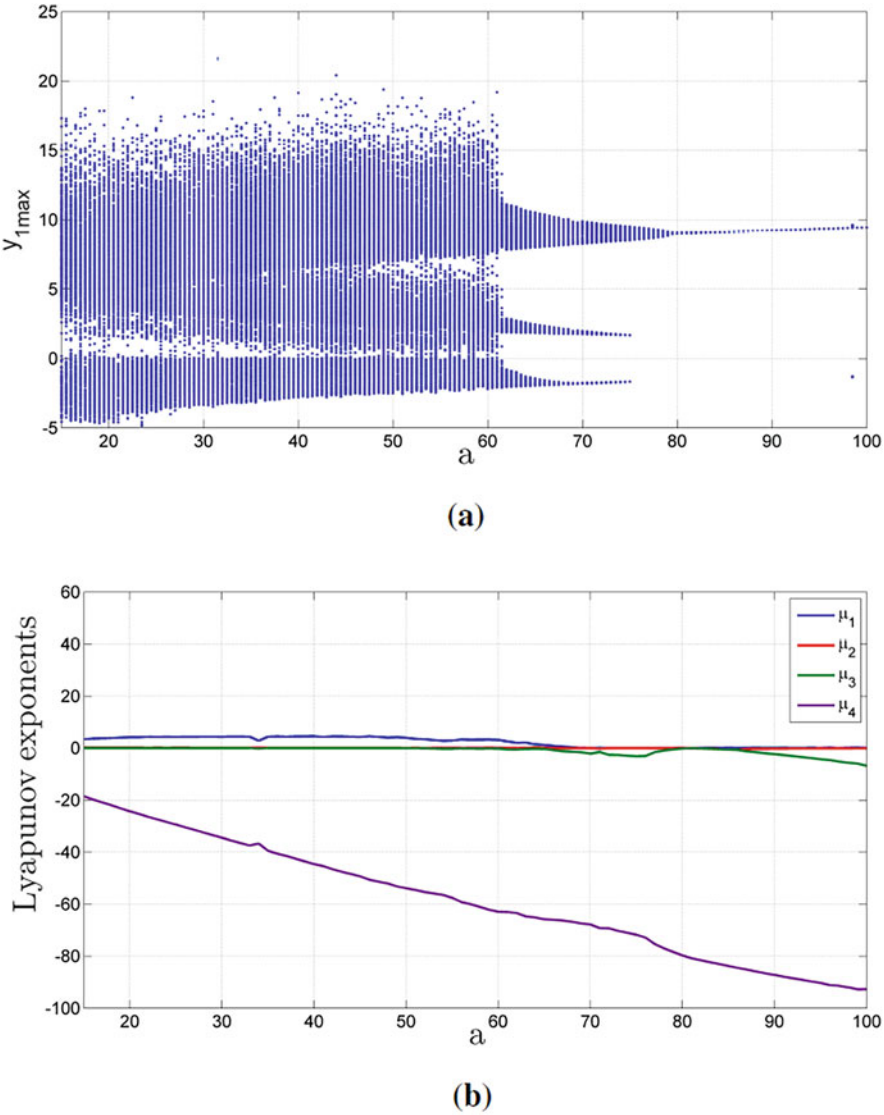


Fig. 9.2 Bifurcation analysis of the system (9.1) when $b = 20$, $c = 2$, and a varies in [15, 100]. (**a**) Bifurcation diagram. (**b**) Lyapunov exponents

By selecting the parameter value as $a = 60$, the chaotic attractor of the system (9.1) in (y_1, y_2) space is plotted in Fig. 9.3b, and its corresponding Lyapunov exponents are obtained as follows:

$$\mu_1 = 3.182,\ \mu_2 = 0,\ \mu_3 = -0.213,\ \mu_4 = -62.936 \tag{9.7}$$

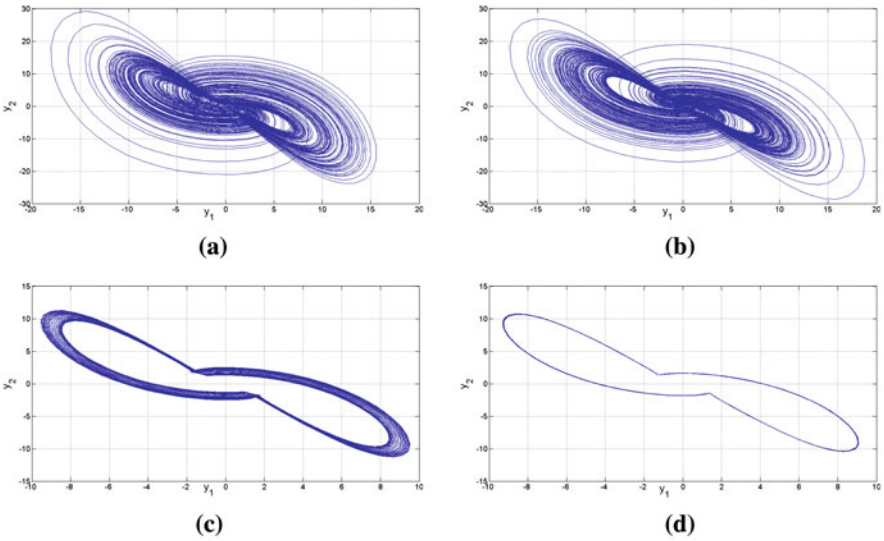


Fig. 9.3 Matlab phase-space plots of the new 4-D hyperchaotic system ((9.1) in the (y_1, y_2) plane, keeping $b = 20$ and $c = 2$, and varying a, with $Y(0) = (0.3, 0.1, 0.3, 0.1)$. (**a**) $a = 40$. (**b**) $a = 60$. (**c**) $a = 75$. (**d**) $a = 90$

In this case, the Lyapunov dimension of (9.1) is derived as

$$D_L = 3 + \frac{\mu_1 + \mu_2 + \mu_3}{|\mu_4|} = 3.0472 \tag{9.8}$$

When $a \in [70, 81]$, and keeping $b = 20$ and $c = 2$, the new 4-D hyperchaotic system (9.1) losses its complexity and its state variables converge to a quasi-periodic attractor, as shown in Fig. 9.3c.

The Lyapunov exponents are calculated for $a = 75$ and obtained as follows:

$$\mu_1 = 0\ \mu_2 = 0,\ \mu_3 = -3.144,\ \mu_4 = -71.779 \tag{9.9}$$

Finally, the system (9.1) displays periodic behavior with no complexity when a increases in the interval [81, 100].

By choosing $a = 90$, and keeping $b = 20$ and $c = 2$, the periodic attractor of the new 4-D hyperchaotic system in (y_1, y_2) plane is plotted in Fig. 9.3d, and its corresponding Lyapunov exponents are obtained as follows:

$$\mu_1 = 0,\ \mu_2 = -0.194,\ \mu_3 = -2.334,\ \mu_4 = -87.242 \tag{9.10}$$

9.3.2 Bifurcation Analysis When b Varies

In the previous subsection, the parameter a was varied while keeping $b = 20$ and $c = 2$. In this subsection, we show the effect of the new 4-D hyperchaotic system (9.1) fixing $a = 15$ and $c = 2$, while varying b in the interval [0.1, 20].

The corresponding bifurcation diagram and Lyapunov exponents' spectrum are obtained numerically by increasing the parameter b from 0.1 to 20 with the initial conditions $Y(0) = (0.3, 0.1, 0.3, 0.1)$. The results are plotted in Fig. 9.4a and b, for the bifurcation and Lyapunov exponents, respectively.

It is observed from Fig. 9.4 that the new 4-D system (9.1) can exhibit hyperchaos for a wide range of values of parameter b. It can also exhibit chaos, quasi-periodic, and periodic behavior for some other intervals. For instance, we will show the dynamics when defining the intervals: $I = [0.1, 0.3] \cup [0.9, 1.5] \cup [8, 8.1]$.

When $b \in I$, and keeping $a = 15$ and $c = 2$, the new 4-D system (9.1) displays periodic behavior with no complexity and its (y_1, y_3) phase-space portrait is depicted in Fig. 9.5a.

When $b = 1.2$, and keeping $a = 15$ and $c = 2$, the corresponding Lyapunov exponents are obtained as follows:

$$\mu_1 = 0,\ \mu_2 = -0.119,\ \mu_3 = -0.318,\ \mu_4 = -14.576 \tag{9.11}$$

When b varies within the range [1.5, 2], and keeping $a = 15$ and $c = 2$, the new 4-D system (9.1) still has no complexity and its state variables converge to a quasi-periodic attractor, as shown in Fig. 9.5b.

When $b = 1.8$, and keeping $a = 15$ and $c = 2$, the corresponding Lyapunov exponents are obtained as follows:

$$\mu_1 = 0,\ \mu_2 = 0,\ \mu_3 = -0.017,\ \mu_4 = -14.986 \tag{9.12}$$

When b increases in the interval [0.3, 0.9], and keeping $a = 15$ and $c = 2$, the new 4-D system (9.1) has only one positive Lyapunov exponent, which means that it generates a hidden chaotic attractor.

By selecting the parameter value as $b = 0.6$, and keeping $a = 15$ and $c = 2$, the chaotic attractor of the system (9.1) is plotted in Fig. 9.5c. The corresponding Lyapunov exponents are obtained as follows:

$$\mu_1 = 0.122,\ \mu_2 = 0,\ \mu_3 = -0.199,\ \mu_4 = -14.922 \tag{9.13}$$

In this case, the Lyapunov dimension of the new 4-D system (9.1) is derived as:

$$D_L = 3 + \frac{\mu_1 + \mu_2 + \mu_3}{|\mu_4|} = 2.9948 \tag{9.14}$$

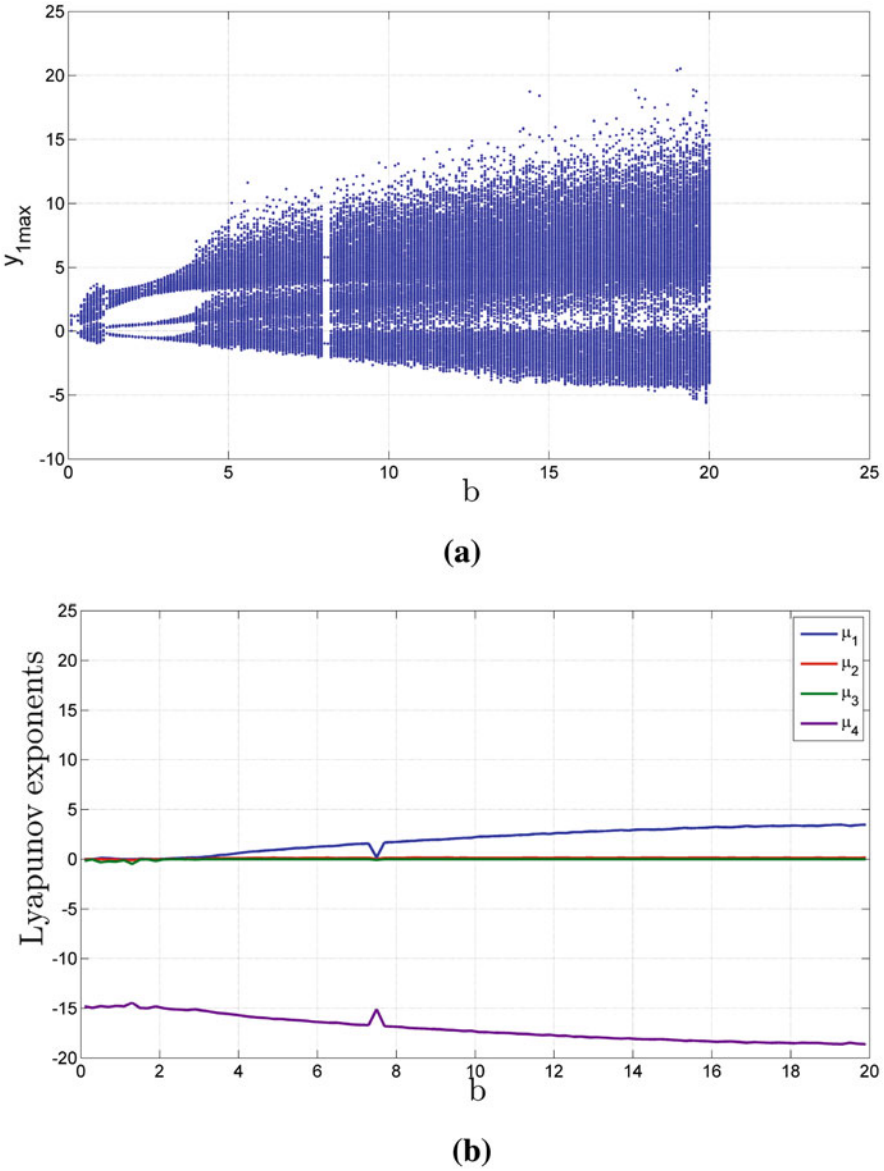


Fig. 9.4 Bifurcation analysis and Lyapunov exponents' spectrum of the new 4-D hyperchaotic system (9.1) when $a = 15$, $c = 2$, and b varies in [0.1, 20]. (**a**) Bifurcation diagram. (**b**) Lyapunov exponents

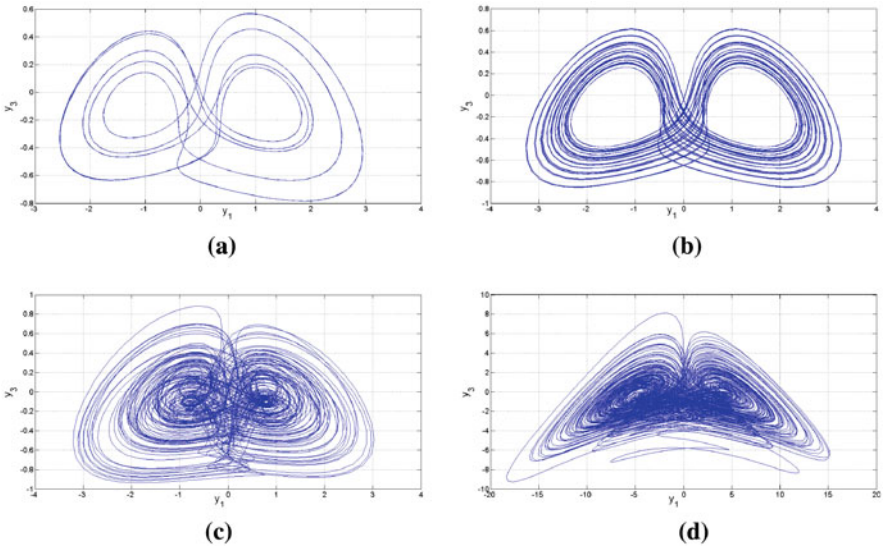


Fig. 9.5 Matlab phase-space plots of the new 4-D system (9.1) in the (y_1, y_3) plane for $a = 15$, $c = 2$, and considering various values of b, solved with $Y(0) = (0.3, 0.1, 0.3, 0.1)$. (**a**) $b = 1.2$. (**b**) $b = 1.8$. (**c**) $b = 0.6$. (**d**) $b = 20$

Finally, when $b \in [2, 20]$, and keeping $a = 15$ and $c = 2$, the system (9.1) displays hidden hyperchaotic behavior with a maximal Lyapunov exponent (MLE) about 3.5, indicating high complexity of the hyperchaotic dynamics of the system (9.1).

Choosing $b = 20$, and keeping $a = 15$ and $c = 2$, the hyperchaotic attractor of the new 4-D system (9.1) is plotted in the (y_1, y_3) phase-space plane, as shown in Fig. 9.5d. The corresponding Lyapunov exponents are

$$\mu_1 = 3.446,\ \mu_2 = 0.149,\ \mu_3 = 0,\ \mu_4 = -15.589 \tag{9.15}$$

In this case, the Lyapunov dimension of the new 4-D system (9.1) is derived as:

$$D_L = 3 + \frac{\mu_1 + \mu_2 + \mu_3}{|\mu_4|} = 3.2306 \tag{9.16}$$

9.3.3 Bifurcation Analysis When c Varies

The new 4-D system (9.1) has three parameters a, b, and c. In this subsection we show the dynamics when fixing $a = 15$ and $b = 20$, while varying c in the interval $[0, 3]$.

The corresponding bifurcation diagram and Lyapunov exponents' spectrum are obtained numerically by increasing the parameter c with $Y(0) = (0.3, 0.1, 0.3, 0.1)$, and the results are plotted in Fig. 9.6a and b, respectively.

When c increases, it is noted from Fig. 9.6 that the new 4-D system (9.1) has always a complex dynamical behavior. The system (9.1) can display hyperchaos with two positive Lyapunov exponents in a large interval of parameter c. In addition, it can evolve into a chaotic attractor with one positive Lyapunov exponent in the first little interval of parameter c.

When $c \in [0, 0.15]$, the new 4-D system (9.1) presents chaotic behavior and generates a hidden chaotic attractor as depicted in Fig. 9.7a. When $c = 0.05$, and keeping $a = 15$ and $b = 20$, the Lyapunov exponents are obtained as:

$$\mu_1 = 3.619,\ \mu_2 = 0, \mu_3 = 0, \mu_4 = -18.619 \tag{9.17}$$

In this case, the Lyapunov dimension of the new 4-D system is obtained as:

$$D_L = 3 + \frac{\mu_1 + \mu_2 + \mu_3}{|\mu_4|} = 3.1944 \tag{9.18}$$

When $c \in [0.15, 3]$, the new 4-D system (9.1) generates a hidden hyperchaotic attractor with a maximal Lyapunov exponent (MLE) about 3.428, as depicted in Fig. 9.7b. When $c = 3$, and keeping $a = 15$ and $b = 20$, the Lyapunov exponents are obtained as follows:

$$\mu_1 = 3.428,\ \mu_2 = 0.219, \mu_3 = 0, \mu_4 = -18.649 \tag{9.19}$$

In this case, the Lyapunov dimension of the new 4-D system is obtained as:

$$D_L = 3 + \frac{\mu_1 + \mu_2 + \mu_3}{|\mu_4|} = 3.1956 \tag{9.20}$$

9.3.4 *Coexistence of Attractors for the New 4-D Hyperchaotic System*

This subsection demonstrates that the complex phenomenon of coexistence of attractors can be observed in our proposed new 4-D system (9.1). Based on the fact that our 4-D model given in (9.1) is invariant under the coordinates transformation $F : (y_1, y_2, y_3, y_4) \mapsto (-y_1, -y_2, y_3, -y_4)$, we have made an appropriate choice of two different initial states that give rise to the appearance of different coexisting attractors in the same phase space for the same values of system parameters. We have chosen the following initial states that are related to two different colors:

$$\begin{aligned} Y_0 &= (0.3, 0.1, 0.3, 0.1) \quad \text{(blue color)} \\ Z_0 &= (-0.3, -0.1, 0.3, -0.1) \quad \text{(red color)} \end{aligned} \tag{9.21}$$

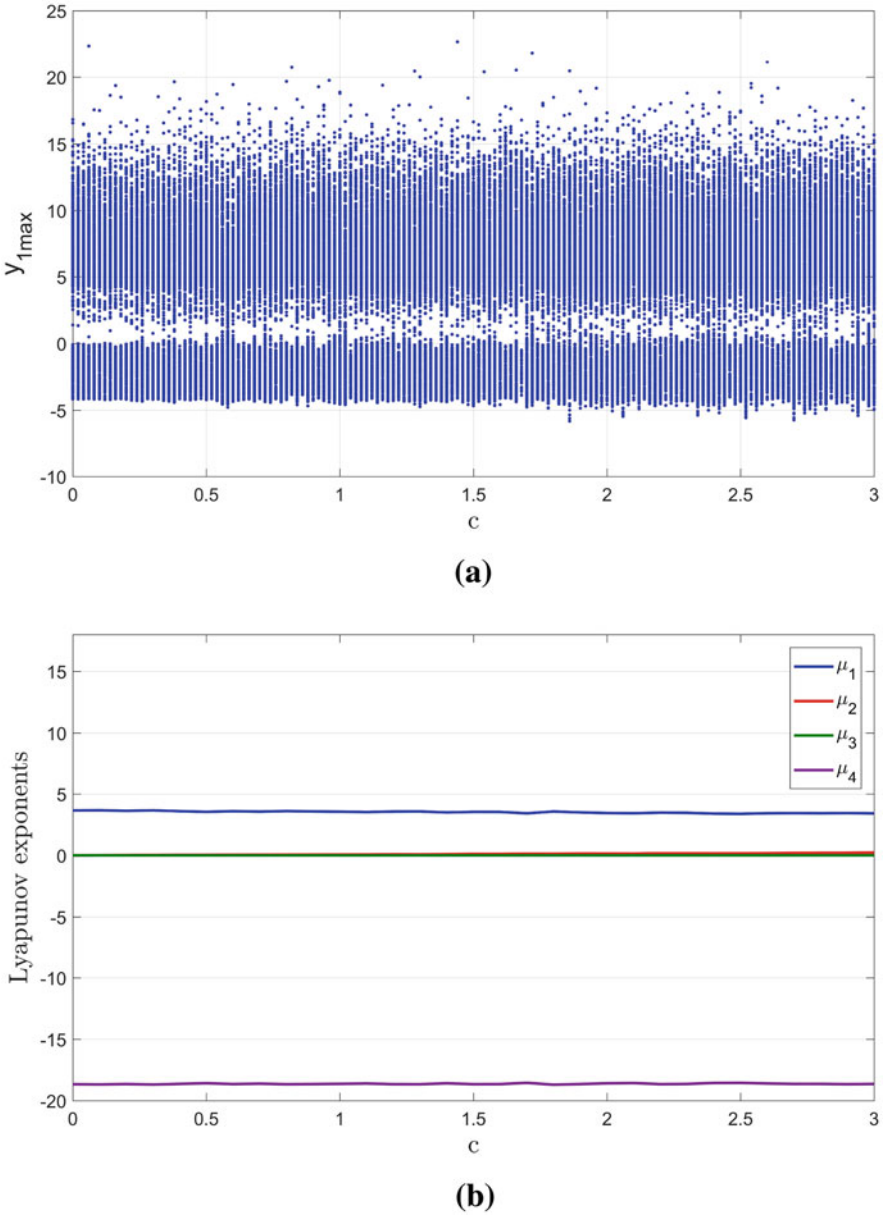


Fig. 9.6 Bifurcation analysis and Lyapunov exponents' spectrum of the new 4-D system (9.1) when $a = 15$ and $b = 20$, and c varies in the range [0, 3]. (**a**) Bifurcation diagram. (**b**) Lyapunov exponents

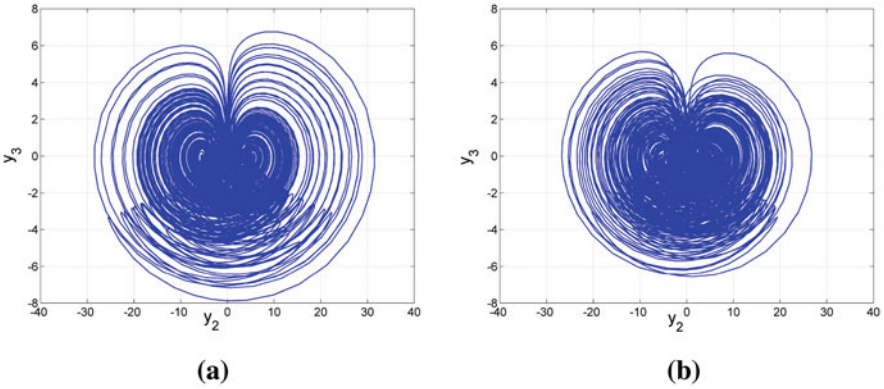


Fig. 9.7 Matlab phase-space plots of the new 4-D system (9.1) in the (y_2, y_3) plane for $a = 15$, $b = 20$, and various values of c, simulated with $Y(0) = (0.3, 0.1, 0.3, 0.1)$. (**a**) $c = 0.05$. (**b**) $c = 3$

In the phase plots shown in Fig. 9.8, the blue phase orbit starts from Y_0, while the red phase orbit starts from Z_0.

When the parameters of the proposed new 4-D system (9.1) are selected as $a = 15$, $b = 0.1$, and $c = 2$, two different coexisting periodic attractors of the system (9.1) can be obtained as depicted in Fig. 9.8a.

When the parameters of the proposed system (9.1) are selected as $a = 15$, $b = 20$, and $c = 0.05$, two different coexisting chaotic attractors of the system (9.1) can be obtained as depicted in Fig. 9.8b.

When the parameters of the proposed system (9.1) are selected as $a = 15$, $b = 20$, and $c = 2$, two different coexisting hyperchaotic attractors of the system (9.1) can be obtained as depicted in Fig. 9.8c.

Finally, coexistence of one periodic attractor starting from Y_0 and one hidden chaotic attractor starting from Z_0 can be obtained when $a = 15$, $b = 0.25$ and $c = 2$ as plotted in Fig. 9.8d.

9.4 MultiSim Electronic Circuit Design of the New 4-D Hyperchaotic System

In this section, an electronic circuit is designed in MultiSim for the new hyperchaotic system (9.1) with no balance point. The electronic circuit is designed using discrete components: resistors, capacitors, operational amplifiers, and multipliers.

In the electronic circuit, the model of the operational amplifier is taken as TL082CD, and the model of the multiplier is taken as AD633JN, with the

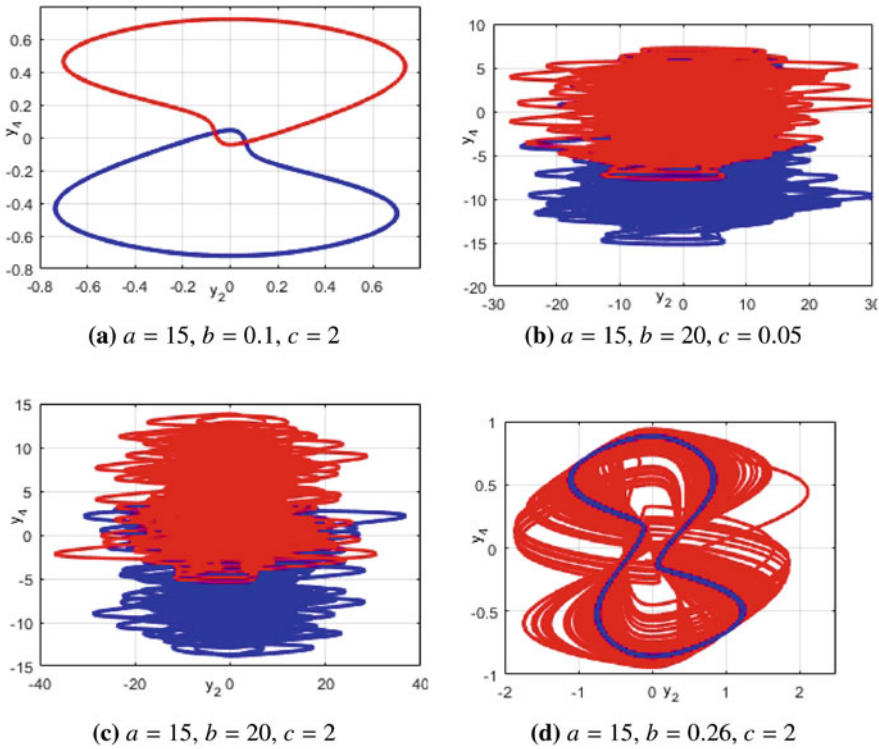


Fig. 9.8 Various coexisting attractors of the proposed new 4-D system (9.1), in the (y_2, y_4) phase-space plane: (**a**) the coexisting periodic attractors, (**b**) the coexisting chaotic attractors, (**c**) the coexisting hyperchaotic attractors, and (**d**) the coexistence of one periodic attractor and one chaotic attractor

multiplication factor of 0.1/V. Applying the Kirchhoff laws, the circuit presented in Fig. 9.9 is described by the following equations:

$$\begin{cases} \dot{y}_1 = -\frac{1}{R_1C_1}\, y_1 - \frac{1}{R_2C_1}\, y_2 + \frac{1}{R_3C_1}\, y_2y_3 + \frac{1}{R_4C_1}\, y_4 \\ \dot{y}_2 = -\frac{1}{R_5C_2}\, y_1 - \frac{1}{R_6C_2}\, y_1y_3 - \frac{1}{R_7C_2}\, y_4 \\ \dot{y}_3 = \frac{1}{R_9C_3}\, V_1 + \frac{1}{R_8C_3}\, y_1y_2 \\ \dot{y}_4 = \frac{1}{R_{10}C_4}\, y_2 \end{cases} \tag{9.22}$$

The values of the circuit elements are given as follows:

$$R1 = R2 = 26.67\,\text{k}\Omega,\ R3 = R5 = R8 = R10 = 400\,\text{k}\Omega, \tag{9.23}$$

$$R11 = R12 = R13 = R14 = R15 = R16 = 100\,\text{k}\Omega, \tag{9.24}$$

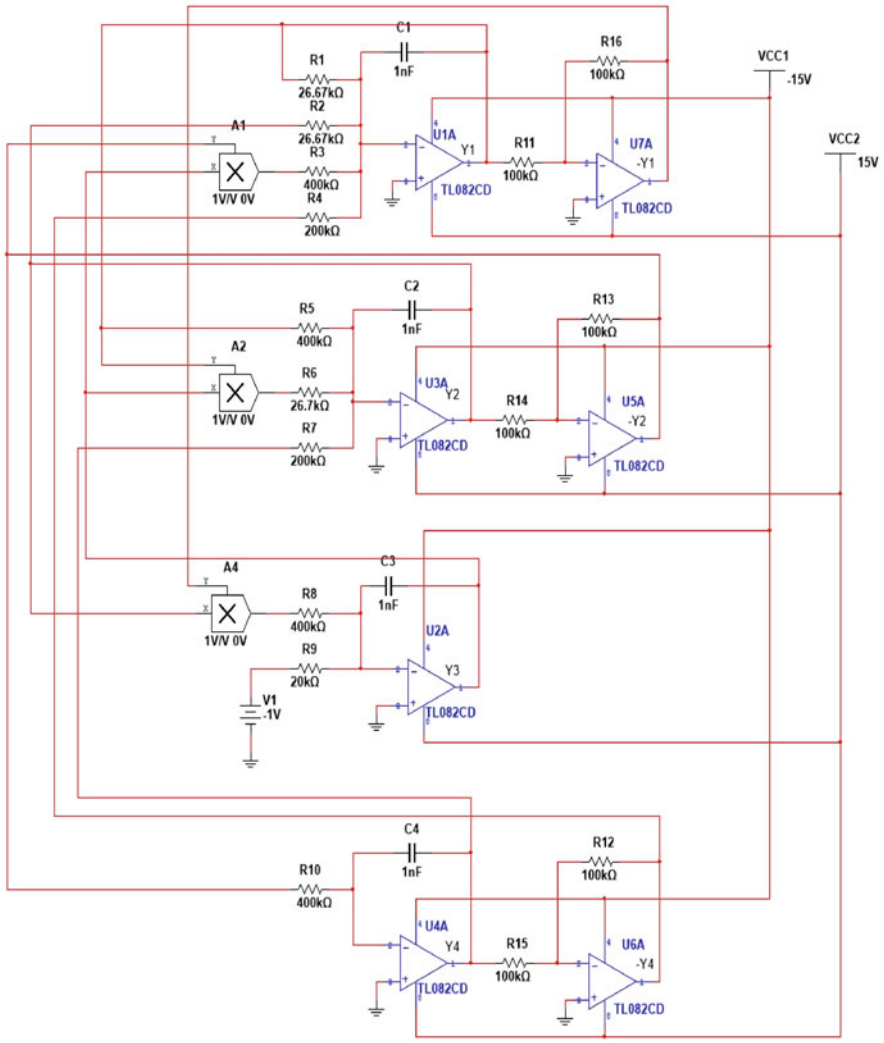


Fig. 9.9 Electronic circuit diagram for the proposed hyperchaotic circuit described by the circuit equations given in (9.22)

$$R4 = R7 = 200\,\mathrm{k}\Omega,\ R6 = 26.67\,\mathrm{k}\Omega,\ R9 = 20\,\mathrm{k}\Omega, \tag{9.25}$$

$$C1 = C2 = C3 = C4 = 1\,\mathrm{nF}. \tag{9.26}$$

Figure 9.10 shows the phase portraits of the electronic hyperchaotic circuit (9.22) simulated by MultiSim 14.0 version, which is consistent with the MATLAB simulation results depicted in Fig. 9.1.

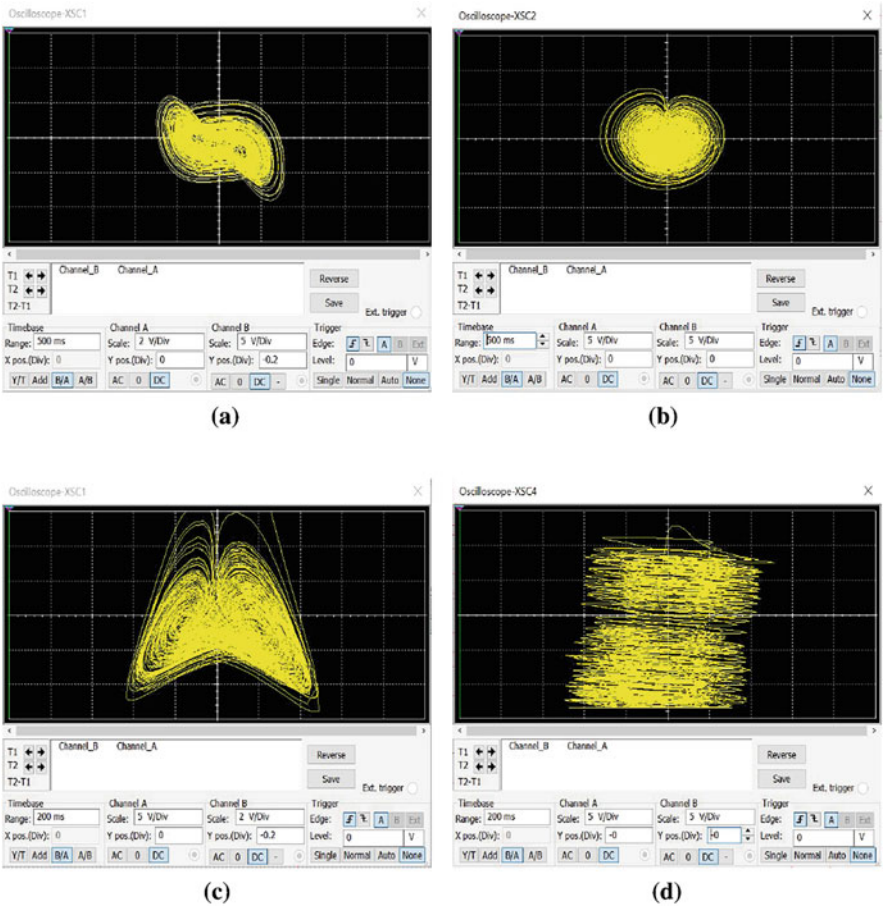


Fig. 9.10 MultiSim results of the new 4-D hyperchaotic circuit (9.22) with no balance point. (**a**) (y_1, y_2) plane. (**b**) (y_2, y_3) plane. (**c**) (y_1, y_3) plane. (**d**) (y_1, y_4) plane

9.5 FPGA Design of the New 4-D Hyperchaotic System with No Balance Point

The new 4D hyperchaotic system with no balance point can be simulated by applying numerical methods, and even optimized as shown in [23], where the associated time execution depends on the estimation of the step size of every numerical method. As already shown in [10], each numerical method provides different discrete equations that can be implemented into a field-programmable gate array (FPGA), which is an excellent option for verification, fast prototyping, and for low development cost applications along with providing good performance.

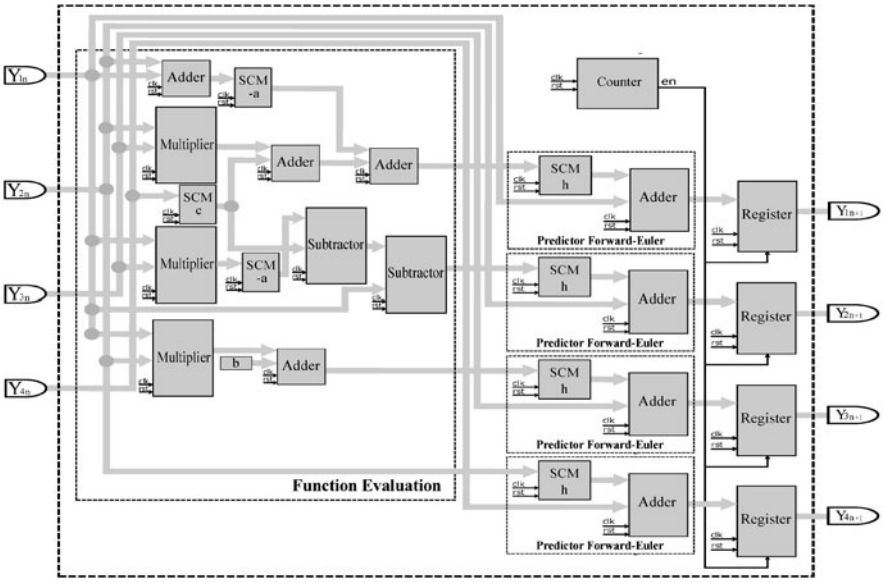


Fig. 9.11 Block diagram description of the new hyperchaotic system (9.1) discretized with Forward-Euler scheme

In this manner, this section shows the FPGA implementation of the new 4-D hyperchaotic system with no balance point given in (9.1). This hyperchaotic system is simulated and implemented by applying three numerical methods, namely: Forward-Euler, Backward-Euler, and fourth-order Runge–Kutta methods. For instance, the Forward-Euler method provides the discrete equations given in (9.27), where h is the step size [23].

$$\begin{aligned} y1_{n+1} &= y1_n + h[-a(y1_n + y2_n) + y2_n y3_n + cy4_n] \\ y2_{n+1} &= y2_n + h[y1_n - ay1_n y3_n - cy4_n] \\ y3_{n+1} &= y3_n + h[b + y1_n y2_n] \\ y4_{n+1} &= y4_n + h[y2_n] \end{aligned} \tag{9.27}$$

In (9.27), one can identify digital blocks that can be implemented in an FPGA, namely: adders, subtractors, and multipliers of two inputs (A,B) and one output (O). Figure 9.11 shows the block diagram description of (9.27), in which the block called single-constant multiplier (SCM) implements the multiplication of a state variable by a constant, such as $ay1_n$, $cy4_n$, and so on. It is worth mentioning that an SCM requires fewer resources and it is faster than a two-input multiplier. The SCM is also designed for all the blocks multiplying h.

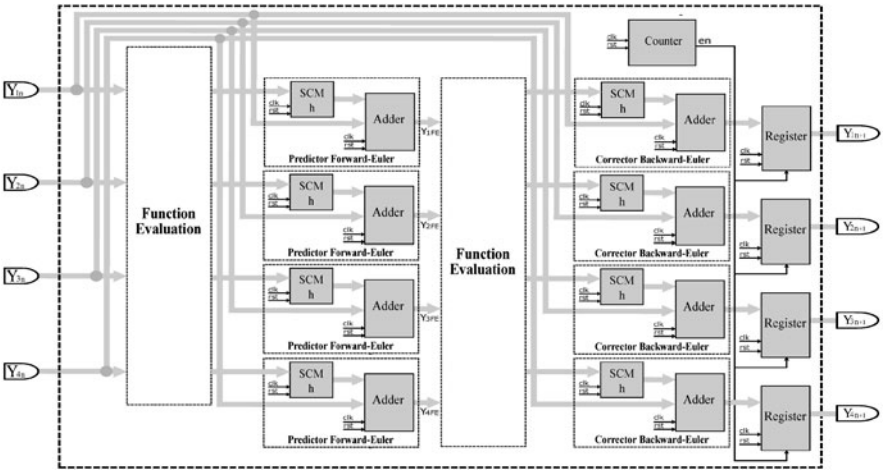


Fig. 9.12 Block diagram description of the new hyperchaotic system (9.1) discretized with Backward-Euler scheme

In an FPGA the blocks can be designed by adopting fixed-point representation with the format 13.19 (32 bits), using one bit for the sign, 12 for the integer part, and 19 for the fractional part. This format in the computer arithmetic to process in the FPGA implementation depends on the amplitudes of the state variables, which can be observed by performing numerical simulations. For example, for this hyperchaotic system, the ranges for y_1 are $[-20, 20]$, for y_2 are $[-35, 35]$, for y_3 are $[-8, 8]$, and for y_4 are $[-25, 25]$. Therefore, the operation $ay_{1_n}y_{3_n}$ will be around ± 2400, so that this value can be represented by using one bit for the sign, 12 bits in the integer part, and 19 bits for the fractional part. The discretization of (9.1) by applying Backward-Euler method can be described by the block diagram shown in Fig. 9.12. The application of the fourth-order Runge–Kutta method will generate a larger block description, but it will provide better accuracy than Forward- and Backward-Euler methods [23].

Table 9.1 shows the hardware resources of the implementation of (9.1) by using the FPGA Cyclone IV EP4CGX150DF31C7 along the synthesizer "Quartus II 13.0." The row called "clock cycles by iteration" represents the number of clock cycles that are required to process a new iteration to compute $y_{1_{n+1}}$, $y_{2_{n+1}}$, $y_{3_{n+1}}$, and $y_{4_{n+1}}$ with a valid data, and the "Latency" row represents the time to compute a new iteration with a 50 MHz clock signal. In all the cases, we set $h = 0.01$.

Table 9.1 Hardware resources for the implementation of (9.1) by using the FPGA Cyclone IV EP4CGX150DF31C7, and by applying the Forward-Euler, Backward-Euler, and fourth-order Runge–Kutta methods with $h = 0.01$

Resources	Forward-Euler	Backward-Euler	Fourth-order Runge–Kutta	Available
Logic elements	1145	2156	4894	149,760
Registers	864	1522	3214	149,760
9×9 bit multipliers	24	48	96	720
Max freq. (MHz)	91.52	91.68	98.56	50
Clock cycles by iteration	10	18	36	–
Latency (ns)	200	360	720	–

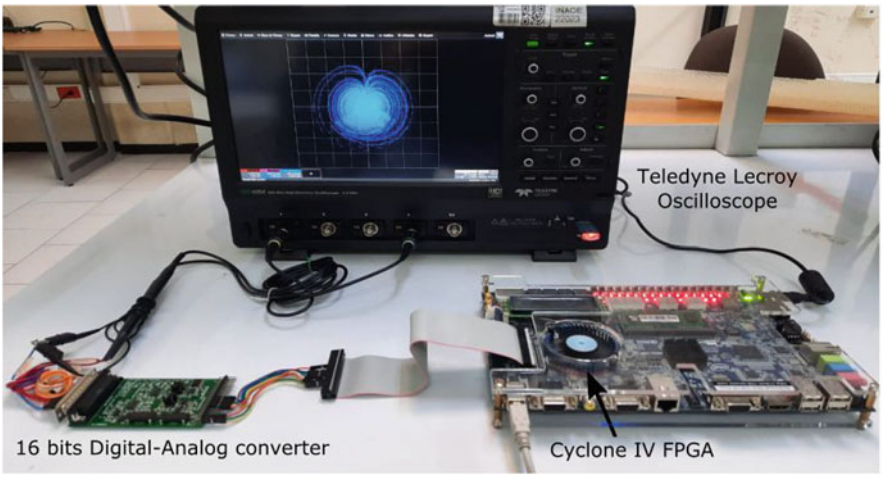


Fig. 9.13 Experimental setup to implement the new 4-D hyperchaotic system (9.1), using a FPGA Cyclone IV EP4CGX150DF31C7, a 16-bit digital-analog converter, and a Teledyne LeCroy oscilloscope to visualize the attractor

Figure 9.13 shows the experimental setup to implement the new hyperchaotic system with no balance point in the FPGA Cyclone IV EP4CGX150DF31C7, and a 16-bit digital–analog converter is used to visualize the results in a Teledyne LeCroy oscilloscope. Figure 9.14 shows the experimental time series generated from the FPGA implementation for the four state variables in (9.1). Figures 9.15, 9.16, and 9.17 show the experimental chaotic attractors of the FPGA implementation of (9.1) by applying the Forward-Euler, Backward-Euler, and the fourth-order Runge–Kutta numerical methods, respectively.

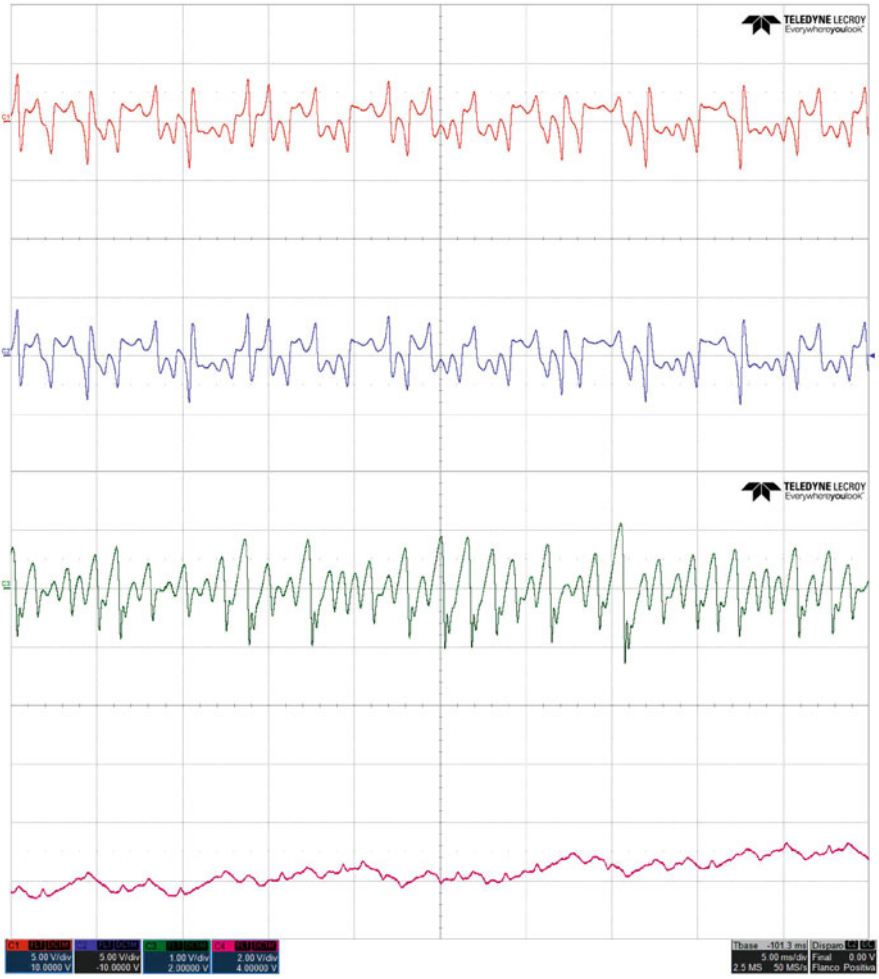
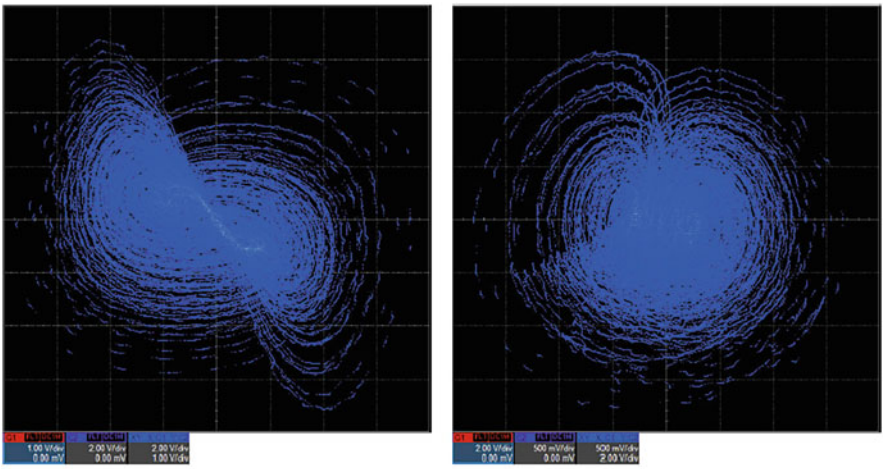


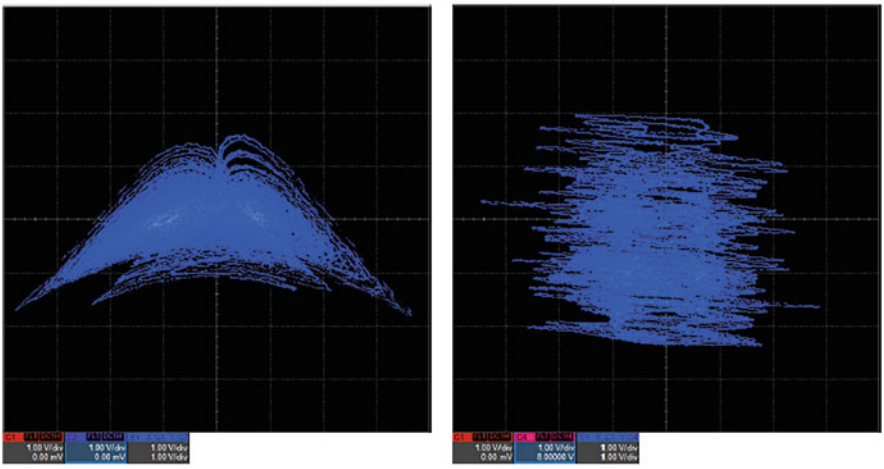
Fig. 9.14 Experimental time series of the new 4-D hyperchaotic system based on (9.1), observed in an oscilloscope, and discretized with Forward Euler with $h = 1 \times 10^{-4}$

9.6 Conclusion

A new 4-D hyperchaotic system with no balance point was proposed in this work and its dynamic properties were analyzed by a bifurcation analysis. We demonstrated that the new system has multi-stability and coexisting attractors. Using MultiSim Version 14, we designed an electronic circuit of the new hyperchaotic system with no balance point. The FPGA implementation of the new 4-D hyperchaotic system with no balance point was performed by applying three numerical methods, and the experimental results were in good agreement to MatLab



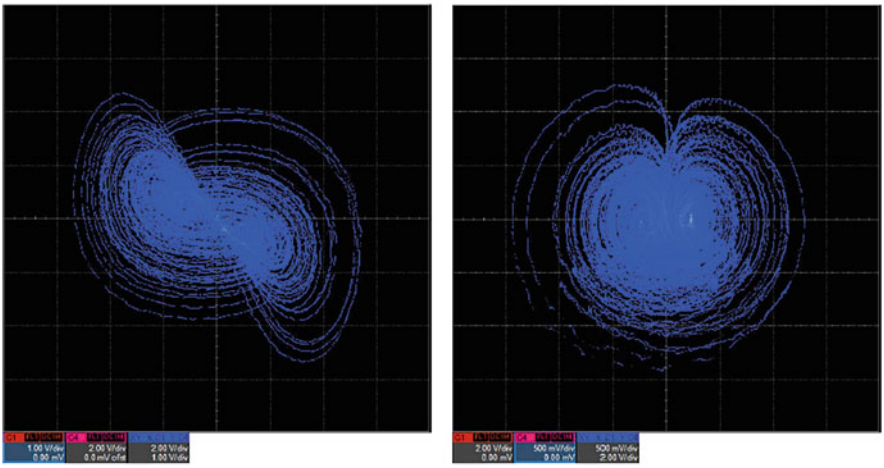
(a) (y_1, y_2) plane (b) (y_2, y_3) plane



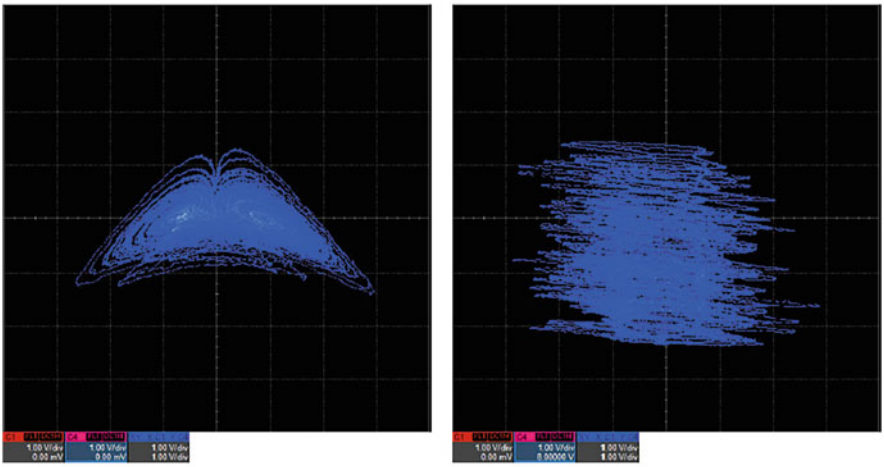
(c) (y_1, y_3) plane (d) (y_1, y_4) plane

Fig. 9.15 Experimental hyperchaotic attractors of (9.1) from the FPGA implementation by applying the Forward-Euler method with $h = 1 \times 10^{-4}$. (**a**) y_1–y_2 view, (**b**) y_2–y_3 view, (**c**) y_1–y_3 view, and (**d**) y_1–y_4 view

simulations. The computer arithmetic for the FPGA implementation was done using 32 bit (1 bit for the sign, 12 bits for the integer part, and 19 for decimal part).

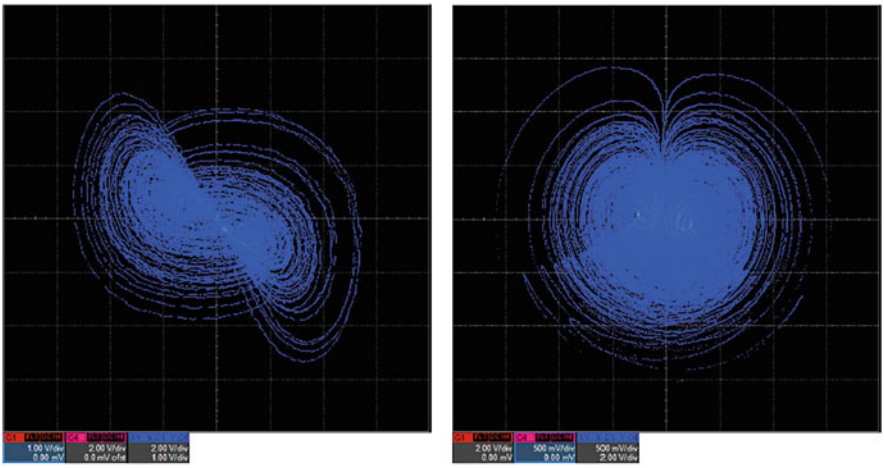


(a) (y_1, y_2) plane (b) (y_2, y_3) plane

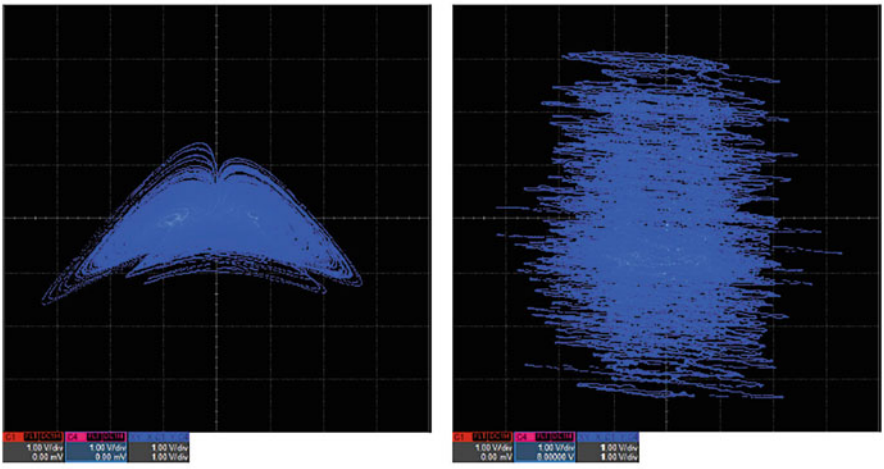


(c) (y_1, y_3) plane (d) (y_1, y_4) plane

Fig. 9.16 Experimental hyperchaotic attractors of (9.1) from the FPGA implementation by applying the Backward-Euler method with $h = 1 \times 10^{-3}$. (**a**) y_1–y_2 view, (**b**) y_2–y_3 view, (**c**) y_1–y_3 view, and (**d**) y_1–y_4 view



(a) (y_1, y_2) plane (b) (y_2, y_3) plane



(c) (y_1, y_3) plane (d) (y_1, y_4) plane

Fig. 9.17 Experimental hyperchaotic attractors of (9.1) from the FPGA implementation by applying the fourth-order Runge–Kutta method with $h = 1 \times 10^{-3}$. (**a**) y_1–y_2 view, (**b**) y_2–y_3 view, (**c**) y_1–y_3 view, and (**d**) y_1–y_4 view

References

1. L. Teng, X. Wang, X. Ye, Hyperchaotic behavior in the novel memristor-based symmetric circuit system. IEEE Access **8**, 151535–151545 (2020)
2. B. Li, X. Zhou, Stabilization of the memristor-based hyperchaotic complex Lü system in a finite time. J. Math. Comput. Sci. **9**(1), 60–72 (2019)

3. H. Wu, X. Hu, Y. Feng, Finite-time synchronisation of memristive hyperchaotic circuit based on Lorenz system with transmission delay. Int. J. Sim. Process Model. **13**(6), 582–588 (2018)
4. E. Barakat, M. Abdel-Aty, I.L. El-Kalla, Hyperchaotic and quasiperiodic behaviors of a two-photon laser with multi-intermediate states. Chaos Solit. Fract. **152**, Article ID 111316 (2021)
5. F.T. Arecchi, Chaos in quantum optics hyperchaos and long memory times. Phys. Scripta. **36**(6), 911–919 (1987)
6. E.K. Petevratzis, C.K. Volos, H.E. Nistazakis, I.N. Stouboulos, I.M. Kyprianidis, An improved motion controller of a mobile robot based on a hyperchaotic system. Int. J. Mech. **12**, 200–204 (2018)
7. M. Boumaraf, F. Merazka, Secure speech coding communication using hyperchaotic key generators for AMR-WB codec. Multimedia Syst. **27**(2), 247–269 (2021)
8. S. Hammami, New PID controller design for multi-switching hyperchaotic synchronization with real-world application. J. Appl. Nonlinear Dyn. **9**(4), 619–642 (2020)
9. E. Tlelo-Cuautle, L.G. De La Fraga, O. Guillén-Fernández, A. Silva-Juárez, *Optimization of Integer/fractional Order Chaotic Systems by Metaheuristics and Their Electronic Realization* (CRC Press, Boca Raton, 2021)
10. E. Tlelo-Cuautle, A.D. Pano-Azucena, O. Guillén-Fernández, A. Silva-Juárez *Analog/Digital Implementation of Fractional Order Chaotic Circuits and Applications* (Springer, Berlin, 2020)
11. J. Gao, M. Wang, X. Peng, Y. Zeng, Compounding self-excited and hidden attractors via a non-autonomous approach. Eur. Phys. J. Special Topics **230**(7–8), 1873–1885 (2021)
12. M. Chen, Z. Wang, F. Nazarimehr, S. Jafari, A novel memristive chaotic system without any equilibrium point. Integration. **79**, 133–142 (2021)
13. Y. Yang, L. Huang, J. Xiang, H. Bao, H. Li, Generating multi-wing hidden attractors with only stable node-foci via non-autonomous approach. Phys. Scripta. **96**(12), Article ID 125220 (2021)
14. S. Mobayen, A. Fekih, S. Vaidyanathan, A. Sambas, Chameleon chaotic systems with quadratic nonlinearities: an adaptive finite-time sliding mode control approach and circuit simulation. IEEE Access **9**, 64458–64573 (2021)
15. J. Llibre, Y. Tian, The zero-Hopf bifurcations of a four-dimensional hyperchaotic system. J. Math. Phy. **62**(5), Article ID 052703 (2021)
16. Y. Deng, Y. Li, Bifurcation and bursting oscillations in 2D non-autonomous discrete memristor-based hyperchaotic map. Chaos Solit. Fract. **150**, Article ID 111064 (2021)
17. I. Ratas, K. Pyragas, P. Tass, Multistability in a star network of Kuramoto-type oscillators with synaptic plasticity. Sci. Rep. **11**(1), Article ID 9840 (2021)
18. Y. Cheng, F. Min, Z. Rui, Y. Dou, Firing multistability, symmetry, bubbles of a Shinriki oscillator with mem-elements. Chinese J. Phy. **74**, 157–174 (2021)
19. J. Ramadoss, J. Kengne, J. Koinfo, K. Rajagopal, Multiple Hopf bifurcations, period-doubling reversals and coexisting attractors for a novel chaotic jerk system with Tchebytchev polynomials. Phys. A Stat. Mech. App. **587**, Article ID 126501 (2022)
20. Y. Bian, W. Yu, A secure communication method based on 6-D hyperchaos and circuit implementation. Telecomm. Syst. **77**(4), 731–751 (2021)
21. M. Gupta, R. Chauhan, Secure image encryption scheme using 4D-Hyperchaotic systems based reconfigurable pseudo-random number generator and S-Box. Integration **81**, 137–159 (2021)
22. S. Vaidyanathan, A. Sambas, E. Tlelo-Cuautle, A.A. Abd El-Latif, B. Abd-El-Atty, O. Guillén-Fernández, K. Benkouider, M.A. Mohamed, M. Mamat, M.A.H. Ibrahim, A new 4-D multi-stable hyperchaotic system with no balance point: Bifurcation analysis, circuit simulation, FPGA realization and image cryptosystem. IEEE Access **9**, 144555–144573 (2021)
23. M.A. Valencia-Ponce, E. Tlelo-Cuautle, L.G. de la Fraga, Estimating the highest time-step in numerical methods to enhance the optimization of chaotic oscillators. Mathematics **9**(16), Article ID 1938 (2021)

Chapter 10
Displacement of Equilibria and n-Double Wing Attractor Generation in the Piecewise Linearized Lorenz System

L. J. Ontanon-Garcia, J. Pena-Ramirez , E. S. Kolosovas-Machuca, R. C. Martínez-Montejano, and C. Soubervielle-Montalvo

Abstract The unstable dissipative systems (UDSs) of type I are described by third-order piecewise linear systems having a real negative eigenvalue and a pair of complex conjugated eigenvalues with positive real part. This type of systems can result in scroll behavior if at least two unstable hyperbolic focus-saddle equilibria appear. However, if more equilibria with the same characteristics are added to the system, the scroll or attractors will displace towards the recently added equilibria. In this chapter the equilibria displacement technique is applied to a linearized version of the Lorenz system that satisfies the UDS type I characteristic. An exponential function can fit the location of the equilibrium points that result in this multi-wing attractor generation, and the position of the commutation surfaces in the z-axis is determined by using linear regression. Ultimately, a function in which the solution generates n-double wing at-tractors that increase size on the z-axis is presented.

L. J. Ontanon-Garcia (✉)
Coordinación Académica Región Altiplano Oeste, Universidad Autónoma de San Luis Potosí, Salinas de Hidalgo, S.L.P, Mexico
e-mail: luis.ontanon@uaslp.mx

J. Pena-Ramirez
CONACYT-CICESE, Applied Physics Division, Center for Scientific Research and Higher Education at Ensenada, Ensenada, B.C., Mexico
e-mail: jpena@cicese.mx

E. S. Kolosovas-Machuca
Coordinación para la Innovación y Aplicación de la Ciencia y la Tecnología (CIACYT)-Facultad de Ciencias, UASLP, San Luis, S.L.P, Mexico
e-mail: samuel.kolosovas@uaslp.mx

R. C. Martínez-Montejano
Unidad Académica Multidisciplinaria Zona Media, UASLP, Rioverde, S.L.P, Mexico
e-mail: roberto.montejano@uaslp.mx

C. Soubervielle-Montalvo
Centro de Investigación y Estudios de Posgrado (CIEP), Facultad de Ingeniería, UASLP, San Luis Potosí, S.L.P., Mexico
e-mail: carlos.soubervielle@uaslp.mx

G. Huerta Cuellar et al. (eds.), *Complex Systems and Their Applications*,
https://doi.org/10.1007/978-3-031-02472-6_10

The results of the system shown here are studied and calculated through numerical analyses and simulations.

Keywords Equilibria displacement · Piecewise linear systems · Attractor generation

10.1 Introduction

The term of unstable dissipative systems (UDSs) of type I and II describe piecewise linear (PWL) dynamical systems with specific equilibria stability, as stated in [1–3]. In this case, unstable systems with saddle-focus equilibria are generated by introducing affine vectors that change values according to the position of the states of the system, resulting in commutation surfaces that when crossed change the location of the equilibria of the system and the direction of their resulting trajectories over time [4]. The idea is that, by the correct positioning of the equilibrium points regarding their eigenvectors, the resulting trajectories instead of being unstable stabilize in scroll trajectories.

Moreover, another interesting behavior is the one reported in [2], where the scroll trajectories can be repeated over space by the proper adjustment of the eigenvalues of the system and the location of the equilibria. Here, the authors proposed an unstable system in $\mathbf{R}^4$ where, with the correct location of the equilibria displaced in the x-axis, the resulting trajectory bounds to a stable dissipative double scroll attractor. And, by continuing the generation of new equilibria, multiscroll attractors form trajectories that are complex enough to be later used as a pseudo-random sequence in a cryptographic scheme [5].

Following the same idea, in [6], the system's trajectory is displaced not only in the x-axis but also in y, z and in space. To do so, the stability of the system must be analyzed in advance, in order to determine the intrinsic dynamics of the eigenvectors and whether the equilibria are pulling or pushing the orbits [4]. An important characteristic to mention regarding these approaches is that they all use UDS of type I, which results in an equal number of scrolls that equilibrium points introduced. On the other hand, some known systems, such as the Chua attractor [7], present a similar double scroll behavior. Nevertheless, instead of presenting two equilibrium points, it stabilizes the dynamics with three equilibrium points with the same eigenvalues. Another example of this can also be seen in [8], and the references therein, where Yalçin et al. implemented a family of scroll grid attractors. In this case, the system is based on a PWL function and the coefficient matrix stability falls under the UDS type I approach since it also presents more equilibria than the number of scrolls on the projection of the system, to achieve the stability of the system. Similar to this case, the works in [9–11] describe methods on the generation of n-scroll attractors. However, regardless of the number of scrolls and equilibria in the system, their location must be considered in advance. Otherwise, the intrinsic dynamic of the eigenvectors may not catch the final trajectory of the system [4].

In this work, and in the same light as [6], we propose a method to generate n-double wing attractors that increase in number toward the z-axis. To achieve this, we use a linearized version of the Lorenz system described by Li et al. [12]. The advantage of this system is that it conserves the three characteristic equilibria of the original Lorenz systems. Still, the parameter c acts now as an amplitude control in the size of the attractor and also forms part of the location of the unstable saddle-focus equilibrium points. In this sense, the displacement equilibria technique is applied to generate solutions of the attractor along the z-axis.

10.2 Linearized Lorenz System

Consider the dynamical system described by:

$$\dot{\mathbf{X}} = \mathbf{F}(\mathbf{X}, \mathbf{P}), \tag{10.1}$$

where $\mathbf{X} \in \mathbf{R}^3$ corresponds to the state vector, $\mathbf{F} : \mathbf{R}^3 \to \mathbf{R}^3$ represents the intrinsic dynamics of the system, and $\mathbf{P} \in \mathbf{R}^3$ stands for the set of parameters, which in general are assumed to be constant. The solution of the system induces into the phase space the flow $\varphi^t(\mathbf{X})|_{(t\in\mathbf{R})}$. For each initial condition $\mathbf{X}_0$ of the system and a specific set of parameters $\mathbf{P}$, the system's solution will be represented as $\varphi^t(\mathbf{X}_0)$.

In this case, we consider the complete linearized Lorenz system proposed in [6], which is described by:

$$\begin{aligned} \dot{x} &= y - x, \\ \dot{y} &= -z \cdot sign(x) + c \cdot sign(x) - a \cdot y, \\ \dot{z} &= x \cdot sign(y) - b \cdot z, \end{aligned} \tag{10.2}$$

where $\mathbf{P} = \{a, b, c\}$ are the parameters and $\mathbf{X} = \{x, y, z\} \in \mathbf{R}^3$ correspond to the state variables, and $sign(\cdot)$ denotes the sign function. A trajectory of the system $\varphi^t(\mathbf{X}_0)$ can be appreciated in Fig. 10.1 for the set of initial conditions $\mathbf{X}_0 = [1, 1, 1]$ and $a = 0.1, b = 0.2$, and $c = 2.8$. Note that for these parameter values, the system oscillates in a chaotic regime. As mentioned by the authors in [12], the system represented in Eq. (10.2) was adapted from the original Lorenz system after a linearization process with the sign functions and an adjustment in size control and sensitivity to the initial conditions. An important characteristic to mention is that the linearized version also presents three equilibrium points $\mathbf{X}^*_{1,2,3}$ as the original Lorenz system. These equilibria are located at:

$$\begin{aligned} \mathbf{X}^*_{1,2} &= [\pm\tfrac{bc}{1+ab}, \pm\tfrac{bc}{1+ab}, \pm\tfrac{c}{1+ab}], \\ \mathbf{X}^*_3 &= [0, 0, 0]. \end{aligned} \tag{10.3}$$

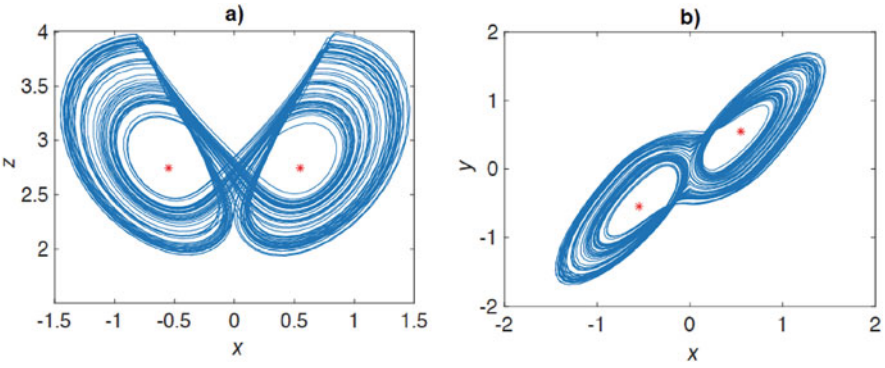


Fig. 10.1 Projection of the linearized Lorenz system given in Eq. (10.2) with $a = 0.1$, $b = 0.2$, and $c = 2.8$ onto the (**a**) (x, z) and (**b**) (x, y) planes. The red asterisks mark the position of the equilibria of the system for the specific set of **P**

Now, in order to determine the stability of the system, the Jacobian matrix will be represented depending on the values of the $sign(\cdot)$ functions in the following way:

$$\mathbf{J} = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -a & -1 \cdot sign(x) \\ 1 \cdot sign(y) & 0 & -b \end{bmatrix}, \tag{10.4}$$

where **J** does not depend on the parameter c. If we consider a and b with the values presented above, there are four possible scenarios for **J** and its corresponding eigenvalues, which are represented as:

(a) If $sign(x) > 0$ and $sign(y) > 0$, then:
$\mathbf{\Lambda}_a = \{-1.5276, 0.1138 \pm 0.8092i\}$.
(b) If $sign(x) < 0$ and $sign(y) > 0$, then:
$\mathbf{\Lambda}_b = \{-0.6339, -0.9669 \pm 0.7816i\}$.
(c) If $sign(x) < 0$ and $sign(y) < 0$, then:
$\mathbf{\Lambda}_c = \{-1.5276, 0.1138 \pm 0.8092i\}$.
(d) If $sign(x) > 0$ and $sign(y) < 0$, then:
$\mathbf{\Lambda}_d = \{-0.6339, -0.9669 \pm 0.7816i\}$.

With $\Lambda_a = \Lambda_c$ and $\Lambda_b = \Lambda_d$, here are two possible eigenvalue sets Λ_a and Λ_b satisfying the UDS definitions of type I stated in [5] and a stable focus node. This means that $X_{1,2}$ present two equilibrium points with unstable saddle-focus type due to Λ_a and the oscillations will be driven around them, as shown in Fig. 10.1a, from which the symmetric behavior due to the identical equilibria (marked in red asterisks) is observed. This behavior is similar to the one observed in UDS type I systems, as presented in [1, 2, 4, 5].

At this point, it is important to remark that the system given in Eq. (10.2) presents three equilibrium points and an attractor with only two scrolls or wings, as it is

appreciated in Fig. 10.1, in contrast to the systems discussed in [1, 2, 4, 5], where they present an equal number of equilibrium points and scrolls. Nevertheless, the displacement technique will only be applied to the saddle-focus equilibria $\mathbf{X}^*_{1,2}$.

10.3 Displacement of the Equilibria of the Linearized Lorenz System

The method of displacing the equilibria proposed in [4, 6] consists of previously studying the stability of the system and then changing the location of the equilibrium points in a particular axis or in space. Here, considering the equilibria in Eq. (10.3), after any variation of the parameter c, the system will only displace $\mathbf{X}^*_{1,2}$ in space, leaving the same position for $\mathbf{X}^*_3$. So, we will analyze this parameter by a bifurcation diagram to understand the dynamics of the system; this is depicted in Fig. 10.2a. In this case, the local maxima points in the z-axis with respect to the x-axis position for a range of values for the parameter c were captured. To understand each vertical plotted set of points for each corresponding value in c, a projection of the local maxima in one particular value is represented in Fig. 10.2b, where the solution for $c = 2.8$ along with the black marks indicating the local maxima in z is represented. The black marks are represented as part of Fig. 10.2a in the black dashed line for the corresponding value of c. These two black dotted columns depicted indicate the place of the symmetric scrolls in the positive and negative regions of the space for the x- and y-axis.

Notice that for each increased value of the parameter c, the size of the attractor starts to expand in size and that the equilibria $\mathbf{X}^*_{1,2}$ (depicted in red dots) are also

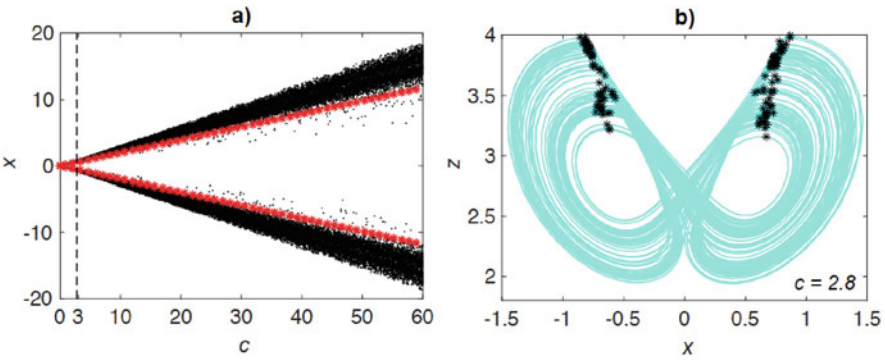


Fig. 10.2 (**a**) Bifurcation diagram of the local maxima of the system given in Eq. (10.2) in the z-axis projected into the x-axis versus the parameter c. (**b**) Projection of the trajectories of system (2) onto the (x, z) plane. The red dots correspond to the location of the corresponding equilibria according to the values of c. The black dots correspond to the location of the local maxima of the solution of the system for each particular value of c (in figure (**b**) only for $c = 2.8$)

separating from each other as the value of c is increasing. The increment that the wings of the attractor are suffering is easy to understand due to the separation of the equilibria, since $\mathbf{X}_1^*$ is separating from $\mathbf{X}_2^*$, and both are separating at the same time from the equilibrium $\mathbf{X}_3^*$, resulting in an attractor with more significant size when the parameter c increases. It is also important to mention that for each assigned value of c the equilibria will result in the same set of eigenvalues as presented before, which means that the dynamics of their eigendirection will not change. Now, following the equilibria location and displacement presented in [6], the idea of this chapter is to design a piecewise value for the parameter c that results in two sets of equilibria that generate the corresponding double wing attractors in space regarding their position. In this case, the following piecewise function for the parameter c will now be defined as:

$$c_{PW} = \begin{cases} c_1 \text{ if } z \leq \alpha_1, \\ c_2 \text{ otherwise.} \end{cases} \tag{10.5}$$

Notice that a switching surface has been generated at $z = \alpha_1$. And now, for every pair of values of c introduced to the piecewise function, the number of switching surfaces will also increase. The values for c_{PW} can be selected arbitrarily for the study depicted ahead. However, we will consider the former one of $c_2 = 2.8$ as one of them. The second parameter value will be selected considering that $c_1 = c_2/2 = 1.4$. To determine the location of the commutation surface α_1, the following analysis was implemented. First, with each individual value of the parameter and the same initial conditions as before, the linearized system given in Eq. (10.2) was simulated. The individual projections of the solutions $\varphi^t(\mathbf{X}_0)|_{c_i}$ with $i = 1, 2$ are represented in Fig. 10.3a, where the orange attractor stands for the solution with $c = c_1 = 1.4$ and the blue attractor for $c = c_2 = 2.8$, both with their respective equilibria in red asterisks.

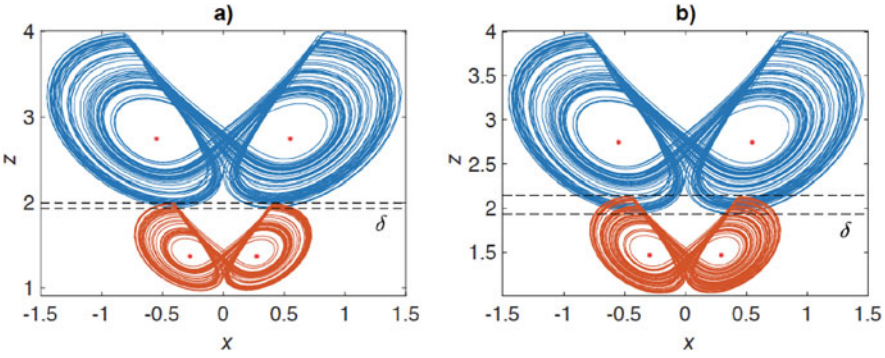


Fig. 10.3 Projections onto the (x, z) plane for the linearized Lorenz system given in Eq. (10.2) for different values of the parameter c. In both graphs, the attractor resulting from a value of $c = 2.8$ is shown in blue, whereas the orange attractor has been generated using (**a**) $c = 1.4$ and (**b**) $c = 1.5$. The asterisks denote the corresponding equilibrium points

Table 10.1 Maximum and minimum values of the solution of the linearized Lorenz system given in Eq. (10.2) for each corresponding value of the parameter C_{PW} in Eq. (10.3). The units are dimensionless

		min_{c_i}			max_{c_i}			$Length_{c_i}$			δ_{c_i}
c_i		x	y	z	x	y	z	x	y	z	% of z
c_1	1.4	−0.7325	−0.85	0.968	0.731	0.849	1.998	1.463	1.70	1.029	6.203
c_2	1.5	−0.7857	−0.91	1.036	0.784	0.911	2.145	1.570	1.824	1.108	19.04
c_3	2.8	−1.4637	−1.70	1.934	1.469	1.707	4.022	2.933	3.408	2.087	–

To understand more about these solutions and the dimensions of their attractors, the maximum $max_{c_i} = max\{\varphi^t(\mathbf{X}_0)|_{c_i}\}$ and $min_{c_i} = min\{\varphi^t(\mathbf{X}_0)|_{c_i}\}$ values that both solutions present with respect to each axis are represented in Table 10.1. These values correspond to each simulated solution for 30,000 iterations using the Runge–Kutta of order 4 (RK4) integration method. Also, in Table 10.1, the column labeled as Length corresponds to the size of the simulated attractor in each axis and is defined as $Length_{c_i} = max_{c_i} - min_{c_i}$.

Now, with these values the ratio of matching locations for the z-axis is calculated in the following way:

$$\delta_{c_i} = \frac{max_{c_i} - min_{c_i}}{Length_{c_i}} \times 100\%. \tag{10.6}$$

This δ_{c_i} represents the percentage of occupation of the attractor in the solution of $\varphi^t(\mathbf{X}_0)|_{c_{i+1}}$ with respect to the length of the solution of $\varphi^t(\mathbf{X}_0)|_{c_i}$ in the z-axis. To understand this, Fig. 10.3a depicts this location represented near $z \approx 2$ labeled δ and marked with two black dashed lines. In this case, the $\max_{c_1} = 1.9982$ and $min_{c_1} = 1.9343$, with the length of the small attractor in the z-axis as $Length_{c_1} = 1.0297$ results in $\delta_{c_1} = 6.2035\%$.

This value indicates that there is only ≈6% of overlapping in the locations in the z-axis for both attractors generated from the corresponding pair of c_1 and c_2 values. Considering that the commutation surface for the parameter c_{PW} in Eq. (10.4) will be chosen within this narrow overlapping space, there is a very small region from which values can be taken for a double wing attractor. Therefore, a change in $c_1 = 1.5$ and the same value in $c_2 = 2.8$ was studied. The projection of this new attractor is depicted in Fig. 10.3b in orange color; the solution of c_2 is again depicted in blue. By analyzing the values of the dimension of the attractors depicted in Table 10.1, now it can be appreciated that δ_{c_1} incremented considerably to 19%, as the region marked with black dashed lines is represented in Fig. 10.3b.

Figure 10.4a and b shows the projections onto the (x, z) and (x, y) planes for the linearized Lorenz system given in Eq. (10.2) for the parameter c_{PW} given in Eq. (10.4) with these values of $c_1 = 1.5$ and $c_2 = 2.8$. The switching surface was chosen as $\alpha_1 = 2$, although for a two-double wing attractor this value can vary as long as it is located between the maximum value of the attractor on the z-axis obtained with c_1 and the minimum value of the second attractor with c_2. The

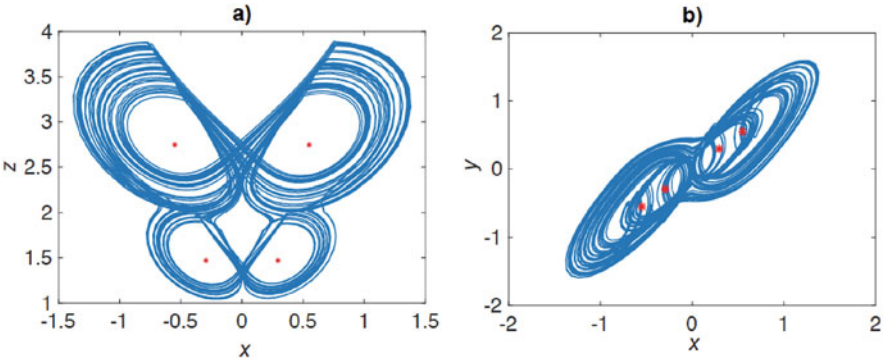


Fig. 10.4 Projection onto the (**a**) (x, z) and (**b**) (x, y) planes for the linearized Lorenz system given in Eq. (10.2) for the parameter c_{PW} given in Eq. (10.4) with $c_1 = 1.5$ and $c_2 = 2.8$ and a switching surface located at $\alpha_1 = 2$. The graph depicts in red asterisk the corresponding equilibria for each c_1 and c_2

graph depicts in blue line the solution of the system $\varphi^t(\mathbf{X}_0)|_{c_{PW}}$ for the same set of initial conditions as before. Notice that the graph depicts a single solution that visits each corresponding domain for the particular values of c_1 and c_2. The system automatically changes the equilibria with respect to the location of the solution in the z-axis. These equilibria are represented in red asterisk for the corresponding values of c_1 and c_2. It is important to mention that if the values of c_1 is adjusted as $c_1 = 1.4$, the flow of the system as it is represented in Fig. 10.4 does not oscillate in this double wing attractor solution, it instead depicts the same solution as for the c_2 in Fig. 10.3 in the blue line.

What follows is extending the generation of these switching surfaces and the equilibria's location for different values of c. For this, we calculated the values of c that match a similar ratio of $19 < \delta_{c_i} < 22$. The idea is to simulate the system given in Eq. (10.2) for a range of c values and measure the ratio of matching locations δ_{c_i} as in Eq. (10.5). The simulation was calculated for the following seven consecutive values $c_1 = 1.5$, $c_2 = 2.8$, $c_3 = 5.2$, $c_4 = 9.6$, $c_5 = 17.7$, $c_6 = 32.5$, and $c_7 = 59.9$. The maximum and minimum values of the attractor on each axis are presented in Table 10.2 along with the corresponding values of δ_{c_i}.

For each pair of equilibria (c_1 and c_2, c_2 and c_3, up to c_6 to c_7), the switching surfaces must be calculated according to the size of the double wing attractors generated for the increasing values of c_1 represented in Table 10.2. The location of the equilibria for these values is represented in Table 10.3. Taking these values into account, the α_i were calculated heuristically; the considerations made were to select values for the position of the switching surface in the z-axis located between the two corresponding equilibria that result in the dynamic of the double wing attractor for the pair of values c_i and c_{i+1} as depicted in Fig. 10.4.

The resulting values for the switching surfaces are: $\alpha_1 = 1.9705$, $\alpha_2 = 3.7450$, $\alpha_3 = 7.0980$, $\alpha_4 = 12.9117$, $\alpha_5 = 23.3529$, and $\alpha_6 = 42.8627$. With these values, a seven-domain region can be designed with the following piecewise parameter:

Table 10.2 Maximum and minimum values of the solution of the linearized Lorenz system given in Eq. (10.2) for each corresponding value of c_i. The units are dimensionless

	min_{c_i}			max_{c_i}			$Length_{c_i}$			δ_{c_i}
c_i	x	y	z	x	y	z	x	y	z	% of z
1.5	−0.79	−0.91	1.04	0.78	0.91	2.15	1.57	1.82	1.11	19.04
2.8	−1.46	−1.70	1.93	1.47	1.71	4.02	2.93	3.41	2.09	20.32
5.2	−2.71	−3.16	3.60	2.72	3.16	7.43	5.43	6.32	3.83	20.33
9.6	−5.00	−5.81	6.65	5.02	5.82	13.70	10.02	11.64	7.05	20.61
17.7	−9.25	−10.7	12.25	9.27	10.77	25.37	18.52	21.52	13.13	21.84
32.5	−16.9	−19.7	22.51	16.95	19.73	46.37	33.91	39.50	23.86	20.36
59.9	−31.3	−36.3	41.51	31.18	36.32	85.39	62.49	72.70	43.88	–

Table 10.3 Equilibria $\mathbf{X}^*_{1,2}$ in Eq. (10.3) of the system given in Eq. (10.2) for the values of the parameter c in Table 10.2. Also, the location of commutation surfaces between each pair of equilibria

c_i	$\pm x^*$	$\pm y^*$	$\pm z^*$	α_j
1.50	0.2941	0.2941	1.4706	1.9706
2.80	0.5490	0.5490	2.7451	3.7451
5.20	1.0196	1.0196	5.0980	7.0980
9.60	1.8824	1.8824	9.4118	12.9118
17.70	3.4706	3.4706	17.3529	23.3529
32.50	6.3725	6.3725	31.8627	42.8627
59.90	11.7451	11.7451	58.7255	–

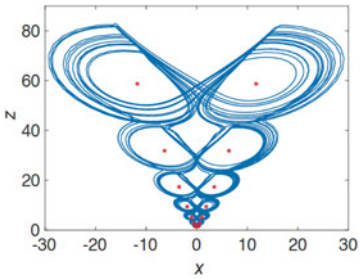


Fig. 10.5 Projection of a seven-double wing attractor of the system given in Eq. (10.2) with the PWL parameter in Eq. (10.5) into the (x, z) plane. The values for c are represented in Table 10.2

$$c_{PW} = \begin{cases} c_1 \text{ if } z \leq \alpha_1, \\ c_2 \text{ if } z \leq \alpha_2, \\ c_3 \text{ if } z \leq \alpha_3, \\ c_4 \text{ if } z \leq \alpha_4, \\ c_5 \text{ if } z \leq \alpha_5, \\ c_6 \text{ if } z \leq \alpha_6, \\ c_7 \text{ otherwise.} \end{cases} \tag{10.7}$$

The projection of the system given in Eq. (10.2) with the parameter from Eq. (10.6) depicts a seven-double wing attractor, as it is shown in Fig. 10.5. The red asterisks represent the location of the equilibria of the system for each generated domain.

10.4 Curve Fitting for the n-Double Wing Attractor Generation

In the previous section we have shown how to generate up to 7-double wing attractors. Here, we elucidate on the growing tendency for the values of c_i and the specific locations for the switching surfaces for the general case of n-double wing attractors. The idea of generating a fitting function for the equilibria displacement is similar as the ones presented in [13] and [14]. First, considering the values of c_i, the increment that they present in Table 10.2 falls on the ≈20% ratio commented in Sect. 10.3 and it seems to fit an exponential curve of the form:

$$\hat{c}_i = \beta_0 e^{(\beta_1 \cdot i)}, \tag{10.8}$$

with $\beta_0, \beta_1 \in \mathbf{R}$, and $i = 1, 2, \ldots, n$ represents the number of switching surfaces introduced by the c_{PW} parameter.

This can be appreciated in the graph of Fig. 10.6a for the marks in red asterisks that correspond to the seven values of c_i in Table 10.2. An exponential curve fitting was implemented with a minimal square linear regression through a logarithmic transformation. The coefficients calculated are $\beta_0 = 0.819106$ and $\beta_1 = 0.613921$ with a coefficient of determination of $r^2 = 0.99998$. The fitted curve is shown in Fig. 10.6a with the blue line. Here, the blue circle marks correspond to the approximated values taken from the model in Eq. (10.7), which result in $\hat{c}_1 = 1.513$, $\hat{c}_2 = 2.796$, $\hat{c}_3 = 5.166$, $\hat{c}_4 = 9.546$, $\hat{c}_5 = 17.639$, $\hat{c}_6 = 32.589$, and $\hat{c}_7 = 60.213$.

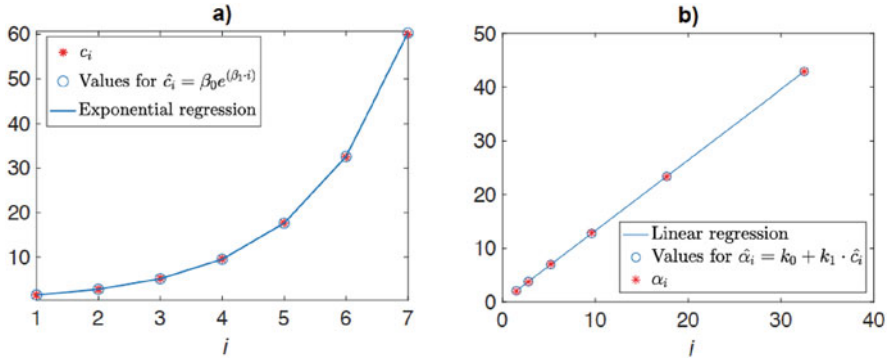


Fig. 10.6 (**a**) Exponential regression for the values of c_i. (**b**) Linear regression for the location of the switching surfaces

Regarding the location of the switching surfaces used in the PWL of Eq. (10.6) (represented in Table 10.3), the location of α_i with respect to the number domains j seems to increment linearly. Therefore, a linear regression fit of the form:

$$\hat{\alpha}_j = k_0 + k_1 \cdot \hat{c}_i \tag{10.9}$$

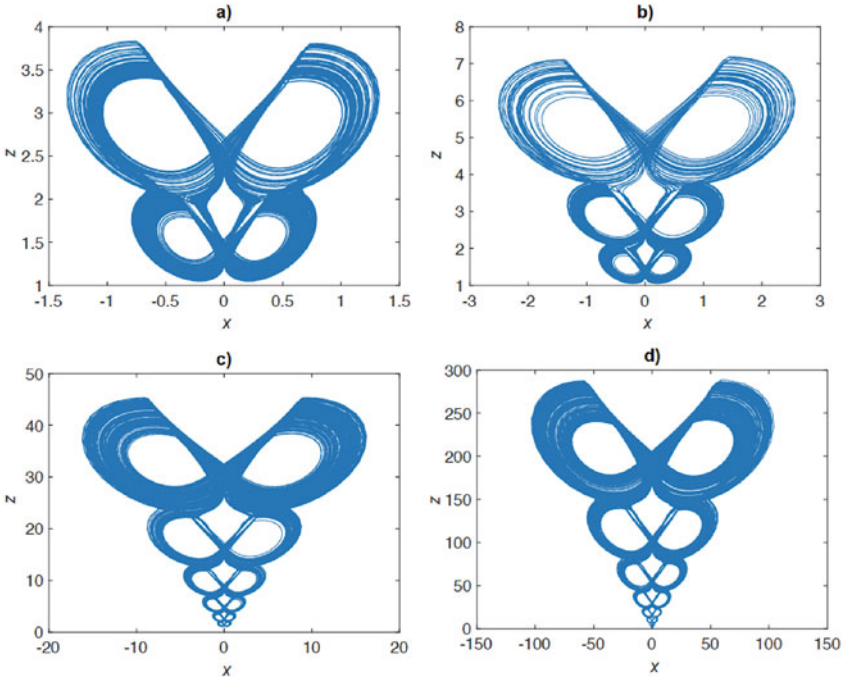
is adjusted with $k_0, k_1 \in \mathbf{R}$ and i stands for the number of switching surfaces and the values of α_i presented for Eq. (10.6). After calculating with a minimal square linear regression, the values result in $k_0 = 0.0594$ and $k_1 = 1.3155$ with a coefficient determination of $r^2 = 0.99994$. The results of this can be seen in Fig. 10.6b, where the values of α_i are depicted in red asterisks, the linear regression $\hat{\alpha}_j$ is represented in the blue line, and the marks in blue circles correspond to the calculated values for $j = 1, 2, \ldots, 6$.

With these two regressions, the design of a n-double wing attractor can be implemented by means of a piecewise function that changes the values of the parameter c according to the exponential curve $\hat{c}_i$ with the switching surfaces located at $\hat{\alpha}_j$. Therefore, the following function is defined:

$$c_{PW} = \begin{cases} \hat{c}_1 & \text{if } z \leq \hat{\alpha}_1, \\ \hat{c}_2 & \text{if } z \leq \hat{\alpha}_2, \\ \vdots & \vdots \\ \hat{c}_{n-1} & \text{if } z \leq \hat{\alpha}_{n-1}, \\ \hat{c}_n & \text{otherwise.} \end{cases} \tag{10.10}$$

With it, any number of double wing attractors can be designed from the system given in Eq. (10.2). Notice that the number of values for the parameter $\hat{c}_i$ must be greater than one than the number of values for the switching surfaces in $\hat{\alpha}_j$. To exemplify this, consider a 2-double, 3-double, 6-double, and 9-double wing attractors with corresponding parameters and switching surfaces $\hat{c}_i, \hat{\alpha}_j$ as it is represented in Fig. 10.7 for the following values: (a) $i = 1, 2$ and $j = 1$; (b) $i = 1, 2, 3$ and $j = 1, 2$; (c) $i = 1, \ldots, 6$ and $j = 1, \ldots, 5$; (d) $i = 1, \ldots, 9$ and $j = 1, \ldots, 8$. The initial condition used was the same as the one represented before. In the same light, a n-double wing attractor can be implemented considering the values of i and j for the system of Eq. (10.2) with the piecewise term in Eq. (10.9).

To determine if the solutions projected in Fig. 10.7 are chaotic, the Lyapunov exponents were calculated using the algorithm proposed by Wolf et al. [15]. The maximum Lyapunov exponent for the four cases is 0.1137, no matter what value of c or the number of double scrolls presented. Therefore, the observed dynamic behavior in all panels of Fig. 10.7 is chaotic.



!ht

Fig. 10.7 Projection onto the (x, z) plane of the system given in Eq. (10.2) with the piecewise parameter from Eq. (10.9) for different number of double wing attractors: (**a**) 2-double, (**b**) 3-double, (**c**) 6-double, and (**d**) 9-double wing attractors

10.5 Conclusions

A method for generating n-double wing attractors for the linearized Lorenz system, which is based on the displacement equilibria technique for UDS type I, has been presented. The study has shown that by switching the value of parameter c, the saddle-focus equilibria of the system can be displaced along the space without changing the corresponding stability. This change of equilibria displaces as well the resulting trajectory of the system along with the scrolls that it presents. Furthermore, it has been shown that the values of the switching parameter c, as well as the location of the switching surfaces α_i can be obtained by using an exponential function and linear regression, respectively. Making it possible to design a fitting function that can be adjusted depending on the number of double scrolls needed. With this proposed function, a 2-, 3,- 6-, and 9-double scroll attractors were generated numerically.

A potential application of the results presented here is in cryptography, as the system presented a similar behavior as in [5]. The results on this matter may be reported elsewhere.

References

1. E. Campos-Cantón, J.G. Barajas-Ramirez,G. Solis-Perales, R. Femat, Multiscroll attractors by switching systems. Chaos Interdiscip. J. Non-linear Sci. **20**(1), 013116 (2010)
2. L.J. Ontanon-Garcia, E. Jiménez-López, E. Campos-Cantón, M. Basin, A family of hyper-chaotic multi-scroll attractors in Rn. Appl. Math. Comput. **233**, 522–533 (2014)
3. R.D.J. Escalante-González, H.E. Gilardi-Velázquez, E. Campos-Cantón, A class of Chua-like systems with only two saddle-foci of different type. IFAC-PapersOnLine **51**(33), 156–161 (2018)
4. L.J. Ontanon-Garcia, E. Campos-Cantón, Widening of the basins of attraction of a multistable switching dynamical system with the location of symmetric equilibria. Nonlinear Analy. Hybrid Syst. **26**, 38–47 (2017)
5. M. García-Martínez, L.J. Ontanon-Garcia, E. Campos-Cantón, S. Čelikovský, Hyperchaotic encryption based on multi-scroll piecewise linear systems. Appl. Math. Comput. **270**, 413–424 (2015)
6. L.J. Ontanon-Garcia, R.E. Lozoya-Ponce, Analog electronic implementation of unstable dissipative systems of type I with multi-scrolls displaced along space. Int. J. Bifurc. Chaos **27**(06), 1750093 (2017)
7. T. Matsumoto, L. Chua, M. Komuro, The double scroll. IEEE Trans. Circuits Syst. **32**(8), 797–818 (1985)
8. M.E. Yalçin, J.A. Suykens, J. Vandewalle, S. Özoğuz, Families of scroll grid attractors. Int. J. Bifurc. Chaos **12**(01), 23–41 (2002)
9. J. Lu, G. Chen, X. Yu, H. Leung, Design and analysis of multiscroll chaotic attractors from saturated function series. IEEE Trans. Circuits Syst. I Regul. Pap. **51**(12), 2476–2490 (2004)
10. S. Yu, J. Lu, H. Leung, G. Chen, Design and implementation of n-scroll chaotic attractors from a general jerk circuit. IEEE Trans. Circuits Syst. I Regul. Pap. **52**(7), 1459–1476 (2005)
11. J. Lu, S. Yu, H. Leung, G. Chen, Experimental verification of multidirectional multiscroll chaotic attractors. IEEE Trans. Circuits Syst. I Regul. Pap. **53**(1), 149–165 (2006)
12. C. Li, J.C. Sprott, W. Thio, Linearization of the Lorenz system. Phys. Lett. A **379**(10–11), 888–893 (2015). https://doi.org/10.1016/j.physleta.2015.01.003
13. H.E. Gilardi-Velázquez, L.J. Ontanon-García, D.G. Hurtado-Rodriguez, E. Campos-Cantón, Multistability in piecewise linear systems versus eigenspectra variation and round function. Int. J. Bifurc. Chaos **27**(09), 1730031 (2017)
14. J.L. Echenausía-Monroy, G. Huerta-Cuéllar, A novel approach to generate attractors with a high number of scrolls. Nonlinear Analy. Hybrid Syst. **35**, 100822 (2020)
15. A. Wolf, J.B. Swift, H.L. Swinney, J.A. Vastano, Determining Lyapunov exponents from a time series. Phys. D Nonlinear Phenomena **16**(3), 285–317 (1985)

Part V
Applications

Chapter 11
Analysis of a Three-Dimensional Non-autonomous Chaotic Circuit with a Thermistor as a Physical Memristor

Laskaridis Lazaros, Volos Christos, and Stouboulos Ioannis

Abstract In 1976, Prof. Leon Chua proposed that a physical thermistor can be modeled as a memristive device, which can be used as a nonlinear element in chaotic circuits. In this direction, an autonomous circuit with two passive elements (inductor and capacitor), a nonlinear resistor, and a thermistor, which plays the role of a nonlinear locally active memristor, has been proposed by Ginoux et al. This work presents the study of a non-autonomous circuit, which is based on the aforementioned autonomous circuit, by adding an external voltage AC source. Moreover, the effect of the capacitor's and inductor's value and the effect of the initial conditions in system's dynamical behavior have been studied. To investigate further system's dynamical behavior, various tools from nonlinear theory have been used, such as bifurcation and maximal Lyapunov exponent diagrams, Poincaré maps, and Kaplan–Yorke dimension. Interesting phenomena related to chaos have been investigated. In more detail, chaotic and regular orbits, such as periodic or semi-periodic, have been observed. Furthermore, the route to chaos through the mechanism of period doubling, coexisting attractors, and crisis phenomena have been observed.

Keywords Non-autonomous circuit · Thermistor · Memristor · Chaos · Coexisting attractors

11.1 Introduction

Leon Chua in 1971 [1] depicted and named the fourth crucial electrical element by finishing a hypothetical group of the other three (resistor, capacitor, and inductor). The name of this element was memristor. A memristor is a non-direct two-terminal electrical element relating electric charge and magnetic flux linkage [2]. In a

L. Lazaros (✉) · V. Christos · S. Ioannis
Laboratory of Nonlinear Systems, Circuits & Complexity (LaNSCom), Department of Physics, Aristotle University of Thessaloniki, Thessaloniki, Greece
e-mail: llaskari@physics.auth.gr; volos@physics.auth.gr; stouboulos@physics.auth.gr

G. Huerta Cuellar et al. (eds.), *Complex Systems and Their Applications*,
https://doi.org/10.1007/978-3-031-02472-6_11

memristor, its resistance decreases when the current flows in a single way and the opposite [3]. At the point when the current stream is halted, memristor holds its last state.

The idea of memristive framework was subsequently summed up by Chua and Kang [4]. Such a framework contains a circuit, of various ordinary elements, which mirrors key properties of the ideal memristor element. The distinguishing proof of memristive properties in electronic elements has drawn in discussion. Tentatively, the ideal memristor is yet to be illustrated [5, 6]. Notwithstanding, a couple of executions interesting circuits have as of late utilized ReRAM memristive models [7–9]. Thus, a physical model of memristor is essential, to understand in depth this fourth circuit element.

Furthermore, while studying the semiconductor conduct of silver sulfide in 1833, Michael Faraday [10] discovered the concept of thermistors. As the temperature rose, he saw that the silver sulfides' opposition decreased. Following it, in 1930, Samuel Ruben invented the basic commercial thermistor [11]. Also, Steinhart and Stanley Hart [12] discovered a capability that thermistor's characteristics have, i.e., the resistance as a function of temperature, which turned out to be appropriate for a wide range of thermistors for ranges of a couple of degrees to two or three hundred degrees. Furthermore, Sah et al. [13] investigated a second-order memristor that depicts the model of a physical device known as a *Positive Temperature Coefficient* and *Negative Temperature Coefficient* thermistor coupled in series.

Thermistors are commonly employed as a linear resistor whose resistance fluctuates with temperature. A negative-temperature coefficient thermistor, in particular, is distinguished by Chua and Kang [4]

$$v_T = R_0(T_0) exp\left[\beta\left(\frac{1}{T} - \frac{1}{T_0}\right)\right] i \triangleq R(T) i, \tag{11.1}$$

where β is the material constant, T is the thermistor temperature, and T_0 is the room temperature both in kelvin. The characteristic curve of the thermistor is modeled with the classical equation of Steinhart–Hart as shown in [14]. The constant $R_0(T_0)$ denotes the cold temperature resistance at $T = T_0$. The instantaneous temperature T is a function of the power dissipated in the thermistor. Moreover, thermistor is governed by the heat transfer equation

$$p(T) = v_T(t) i(t) = \delta(T - T_0) + c\frac{dT}{dt}, \tag{11.2}$$

where c is the heat capacitance and δ is the dissipation constant of the thermistor.

By combining Eqs. (11.1) and (11.2), it is obtained

$$\frac{dT}{dt} = -\frac{\delta}{c}(T - T_0) + \frac{R_0(T_0)}{c} exp\left[\beta\left(\frac{1}{T} - \frac{1}{T_0}\right)\right] i^2. \tag{11.3}$$

As a result of Eq. (11.3), the thermistor is a first-order time-invariant current-controlled memristive. In this study, an autonomous three-dimensional system [14] with a thermistor was converted to a non-autonomous system by introducing an external alternating current source and a linear resistor into the model.

In this chapter, a detailed investigation of the dynamical behavior of the proposed non-autonomous circuit for different values of the capacitance of the capacitor, as well as the inductance of the inductor and the initial conditions, is presented. The capacitance is studied in the region of 0 and 2 F. Also, the inductance belongs between 0 and 15 H. This work is based on the simulation results, which are produced by using well-known numerical tools, such as Lyapunov exponents [15, 16], bifurcation diagrams [17], and Poincaré section. The calculation of the bifurcation diagrams is performed by computing the Poincaré map of the system. The sampling of the Poincaré map is done with the time being an integer multiple of the external period of the system, excluding the transient points. Also, the Lyapunov exponents are computed based on the algorithm from Sandri's [18] package in Mathematica.

The work is organized as follows. In Sect. 11.2, the proposed circuit and its properties are introduced. In Sect. 11.3, the numerical investigation of the circuit's dynamics is presented. Finally, the conclusions of this work are discussed in Sect. 11.4.

11.2 The Proposed Nonlinear Circuit

The proposed framework depends on the independent Muthuswamy–Chua–Ginoux circuit [14], which is changed over to a non-independent circuit by utilizing an external voltage source. The framework comprises a resistor R, a capacitor of capacitance C, an inductor of inductance L, a nonlinear resistor N_R, a thermistor that is a nonlinear locally active memristor M, and an external AC voltage source. The previously mentioned circuit is displayed in Fig. 11.1.

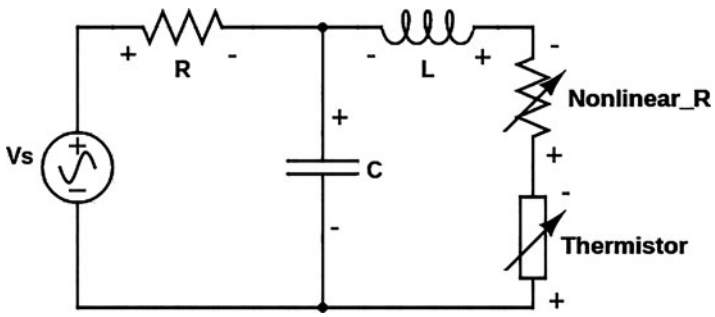


Fig. 11.1 Non-autonomous converted Muthuswamy–Chua–Ginoux circuit

The nonlinear resistor is modeled [14] as a cubic function of the current and it is given by $f(i) = \alpha i + bi^3$. Also as shown in [14] by using Kirchhoff's law for voltages in the left and right loop of the system, the following equations for the voltages are produced:

$$V_S = V_R + V_C \tag{11.4}$$

$$V_C + V_L + V_{N_R} + V_M = 0, \tag{11.5}$$

where V_L is the voltage of the inductor and it is given by $V_L = L\frac{di_L}{dt}$.

In Eq. (11.4), $V_S = V_0 \cos(2\pi f t)$ is the external ac voltage, V_R is the voltage of the linear resistance, and V_C is the voltage of the capacitor.

The voltage of the nonlinear resistor is modeled as a cubic function, and it is given by

$$V_{N_R} = f(i_L) = \alpha i + bi^3, \tag{11.6}$$

where α and b are constants. The voltage of the thermistor is given by Ohm's law and its equation is $V_M = R(T)i_L$. By taking Eq. (11.4) into account and since the current from the capacitor is given by $i_C = C\frac{dV_C}{dt}$ and also from the equation of the currents which is Kirchhoff's first law $i_R = i_L + i_C$, the equations of the system are obtained.

$$\begin{aligned}
\frac{dV_c}{dt} &= \frac{V_S}{RC} - \frac{i_L}{C} - \frac{V_c}{RC} \\
\frac{di_L}{dt} &= -\frac{1}{L}\left(V_c + f(i_L) + R(T)i_L\right) \\
\frac{dT}{dt} &= \frac{R(T)}{c} i_L^2 - \frac{\delta}{c}(T - T_0).
\end{aligned} \tag{11.7}$$

By using the following approximation [14], $R(T)$ is given:

$$R(T) = R_0 \left[1 - \frac{\beta}{T_0^2}(T - T_0) + \frac{\beta(\beta + 2T_0)}{2T_0^4}(T - T_0)^2 + O((T - T_0)^3)\right]. \tag{11.8}$$

To work on the investigation of system (11.7), a change to system's variable is made as

$$x = V_C, y = i_L, z = T - T_0.$$

Also, the following changes to equation's (11.8) parameters have been used.

$$\theta = \frac{R_0}{c}, \gamma = -\frac{R_0}{c}\frac{\beta}{T_0^2}, \mu = \frac{R_0}{c}\frac{\beta(\beta + 2T_0)}{2T_0^4}, \epsilon = \frac{\delta}{c}.$$

Finally, the set of Eqs. (11.7) of system has been transformed to the following set of equations:

$$\begin{aligned} \frac{dx}{dt} &= \frac{V_s}{RC} - \frac{x}{RC} - \frac{y}{C} \\ \frac{dy}{dt} &= -\frac{1}{L}(x + f(y) + R(z)y) \\ \frac{dz}{dt} &= R(z)y^2 - \epsilon z, \end{aligned} \tag{11.9}$$

where $f(y) = \alpha y + by^3$ and $R(z) = \mu z^2 + \gamma z + \theta$. The basic difference of this system, from the system of equation (4) in [14], is the dependence of the time and the existence of more elements in the circuit. Also the units of the variables are in S.I. and especially the capacitance in F, the resistance in Ω, the voltage in V, the frequency in Hz, and the inductance in H.

11.3 Numerical Results

In this section, the dynamical behavior of the proposed, non-autonomous, system (11.9) with $V_0 \neq 0$, $C \neq 0$, and $R \neq 0$ for different values of the capacitance of the capacitor C, the inductance L of the inductor, and the initial conditions is investigated. Generally, the system has rich dynamics that include regular (periodic and semi-periodic) and chaotic oscillations. Also, small changes in the inductance L and in the initial conditions produce a shift between chaotic and regular oscillations and the existence of coexisting attractors, respectively.

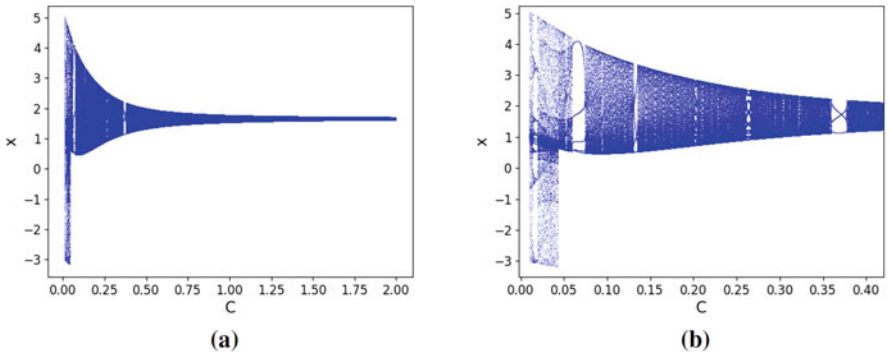


Fig. 11.2 (**a**) Bifurcation diagram of x versus the capacitance C and (**b**) zoom in specific region

11.3.1 The Dynamics Related to the Capacitance C

Figure 11.2 presents the bifurcation diagram of the x variable, which is the voltage of the capacitor, in regard to the capacitance of the capacitor C. Moreover, the values of parameters of the system are $V_0 = 1.0\,\text{V}$, $R = 5\,\Omega$, $f = 0.5\,\text{Hz}$, $\alpha = -6\,\Omega$, $b = 3\frac{\Omega}{A^2}$, $L = 12.2\,\text{H}$, $\mu = 3\frac{\Omega\text{kg}}{\text{J K}}$, $\gamma = -2\frac{\Omega\text{kg}}{\text{J}}$, $\theta = 3\frac{\Omega\text{kg K}}{\text{J}}$, and $\epsilon = 0.6\frac{\text{kg K}}{\text{J}}$. The parameters have been set to these values for two reasons. The first reason is because of the exponential behavior of $R(T)$. More specifically, the resistance of the thermistor $R(T)$ is supposed to be positive, and as a consequence, the right hand of Eq. (11.8) must be positive. The second reason is to find chaotic behavior. Also, the initial conditions are $x_0 = 0.01$, $y_0 = 0$, and $z = 0$. From this bifurcation diagram, a rich dynamical behavior of the system is investigated in regard to the capacitance C. There are regions where the system oscillates chaotically and regions where the system oscillates regularly. In more detail, system's dynamic behavior is chaotic for $C < 0.06\,\text{F}$. Then, the system goes to regular behavior (periodic) and finally to semi-periodic behavior. The maximal Lyapunov exponent diagram verifies this rich dynamical behavior and indicates that after the value of the capacitance ($C = 0.38\,\text{F}$) the system goes to semi-periodic behavior for all the range of the bifurcation parameter. Figure 11.3 presents the maximal Lyapunov exponent, where when the exponent is positive, that means the existence of chaotic behavior (chaotic oscillations) and when it is not positive, that means the system has a regular behavior (periodic and semi-periodic oscillations).

By taking a value of the capacitance ($C = 1.0\,\text{F}$), the system is solved, and the time series of signals x, y and z and the respective phase portrait are presented in Fig. 11.4. Thus, from time series and from the phase, the portrait is observed that the orbit fills densely. So the conclusion is that the orbit is semi-periodic, which is also confirmed from the Poincaré section and maximal Lyapunov exponent of

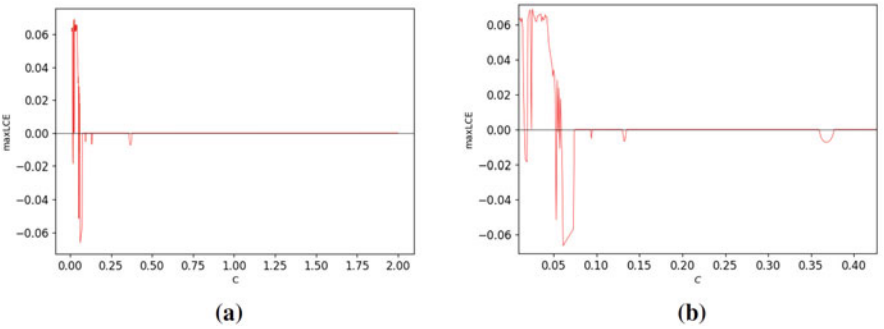


Fig. 11.3 (**a**) Maximal Lyapunov exponent diagram and (**b**) zoom in specific region

Fig. 11.5 where the maximal Lyapunov exponent is equal to $mLCE = 0.000002$. Also, the whole Lyapunov spectrum is $(0.000002, 0, -0.0539074)$. Moreover, the Kaplan–Yorke conjecture [18–20] calculated from equation

$$D = j + \frac{\sum_{i=1}^{j} \lambda_i}{|\lambda_{j+1}|}, \tag{11.10}$$

where λ_i are the Lyapunov exponents, is equal to $D = 2.0$. This means that it is a torus of dimension 2.

11.3.2 The Dynamics Related to the Inductance L

In this section, the numerical results from the simulations regarding the value of inductance L are presented. The bifurcation diagram in regard to the parameter L for specific values of the capacitance of the capacitor C and the amplitude V_0 of the AC voltage source has been produced. In Fig. 11.6, the bifurcation diagram is presented in regard to the inductance L for $C = 0.01\,\text{F}$ and $V_0 = 1.0\,\text{V}$, and in Fig. 11.7, the diagram of maximal Lyapunov exponent is depicted. From the bifurcation diagram of Fig. 11.6, it is observed that the dynamical behavior is changing between chaotic and regular as the inductance L increases, through period-doubling routes. This behavior is also confirmed from the maximal Lyapunov diagram of Fig. 11.7, where the maximal Lyapunov exponent is positive in chaotic regions and no positive in non-chaotic regions. Also, in Fig. 11.8, it is presented the time response of signal x, the phase portraits, and the Poincaré section of the system for $L = 5\,\text{H}$ and $L = 4\,\text{H}$. It is observed that for $L = 5\,\text{H}$ the system has chaotic behavior, and for $L = 4\,\text{H}$, the system's behavior is regular and especially periodic with period 2.

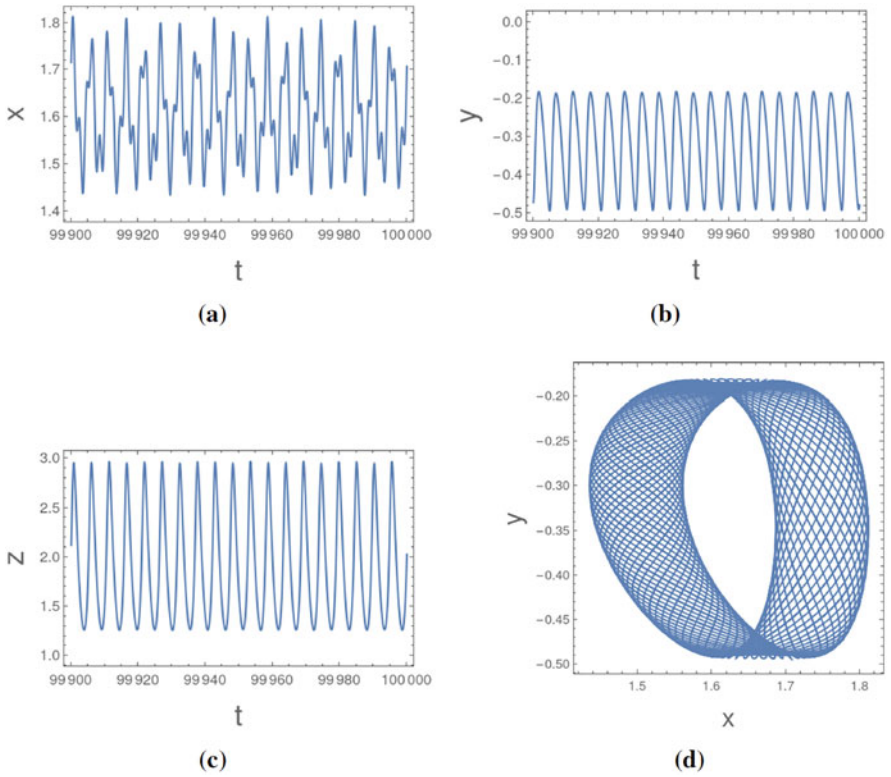


Fig. 11.4 (**a**), (**b**), (**c**) Time response for *x*,*y* and *z* variables and (**d**) phase portrait of system in *x*–*y* plane for $V_0 = 1.0\,\mathrm{V}$ and $C = 1.0\,\mathrm{F}$

11.3.3 The Dynamics Related to the Initial Conditions x_0, y_0, z_0

In this subsection, the dynamical behavior of the system is investigated in regard to the initial condition x_0, y_0, z_0. More specifically, the parameters of the system are $V_0 = 1.0\,\mathrm{V}, C = 0.01\,\mathrm{F}$, and $L = 12.2\,\mathrm{H}$, and now the linear resistance is changed to a higher value and specially to $R = 31\,\Omega$. So, in Fig. 11.9, a bifurcation-like diagram and the maximal Lyapunov exponent diagram in regard to the initial conditions x_0, y_0, and z_0 have been produced.

From the bifurcation-like diagram and the maximal Lyapunov exponent diagram, it is observed the dynamical behavior of the system changes in regard to the initial conditions x_0, y_0, and z_0. More specifically, there are regions where the behavior is only chaotic ($2.7 < x_0 < 5.5$) and regions where the behavior is only regular

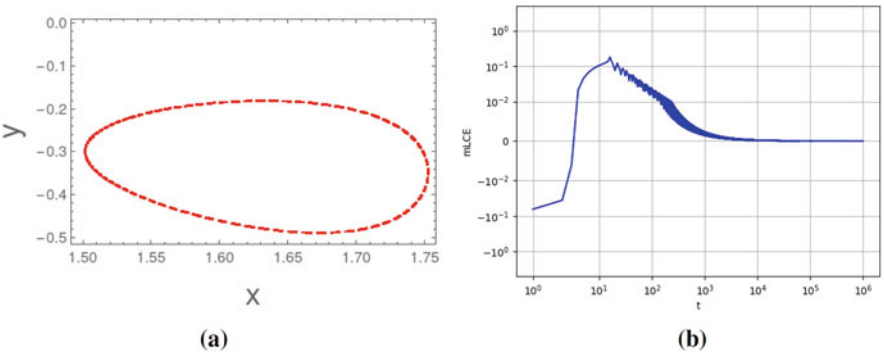


Fig. 11.5 (**a**) Poincaré section and (**b**) maximal Lyapunov exponent of system for $V_0 = 1.0\,\text{V}$ and $C = 1.0\,\text{F}$

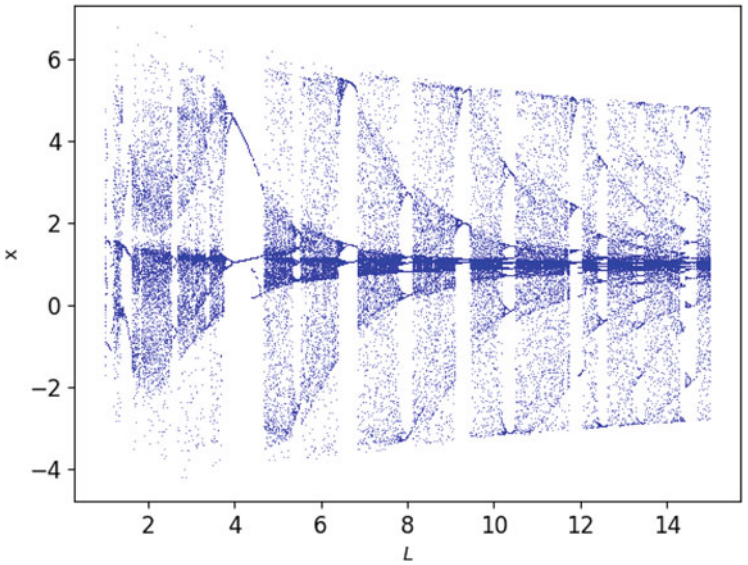


Fig. 11.6 Bifurcation diagram of system in regard to L for $C = 0.01\,\text{F}$

$(7.7 < x_0 < 8.35)$. Also, except from these two regions, the behavior of the system is changing rapidly between chaotic and regular behavior, as it is observed from the bifurcation-like and maximal Lyapunov diagrams. Therefore, the existence of coexisting attractors for different initial conditions is observed. In Fig. 11.10, the phase portraits for $x_0 = 0.06$ (chaotic behavior) and for $x_0 = 8.0$ (regular behavior) are presented.

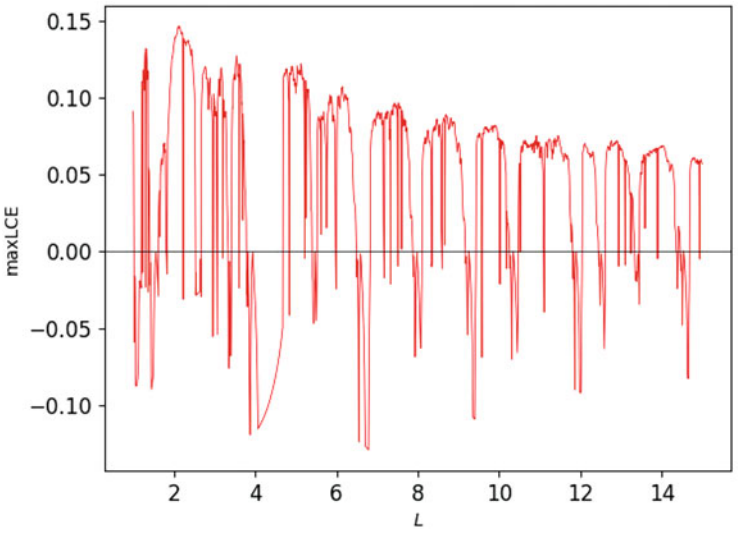


Fig. 11.7 Maximal Lyapunov diagram of system in regard to L for $C = 0.01\,\mathrm{F}$

11.4 Conclusion

A three-dimensional non-autonomous chaotic circuit based on a physical memristor is examined, with a thermistor serving as a "local active" memristor. The thought was to consider the dynamical behavior of the autonomous system by embedding an external AC voltage source and a linear resistance R. Plenty of numerical tools to study the dynamical behavior, such as bifurcation and bifurcation-like diagrams, diagrams of maximal Lyapunov exponent, and the Poincaré map, were used.

The non-autonomous system (11.9) presented rich dynamical behavior. Chaotic and regular behavior were observed. More specifically, chaotic, periodic, and semi-periodic orbits were revealed. Moreover, the system presented route to chaos through the mechanism of period doubling as well as crisis phenomena. From bifurcation diagrams in regard to the capacitance C, it is observed that as the capacitance increases from $C = 0.01\,\mathrm{F}$ to $C = 2.0\,\mathrm{F}$, the dynamical behavior of the system becomes regular (semi-periodic) in all the range as it is presented.

The second approach to our system was to change the inductance L and study the dynamical behavior. In this case, it is observed that, as the inductance increases, the dynamical behavior is changing between regular and chaotic as shown from the

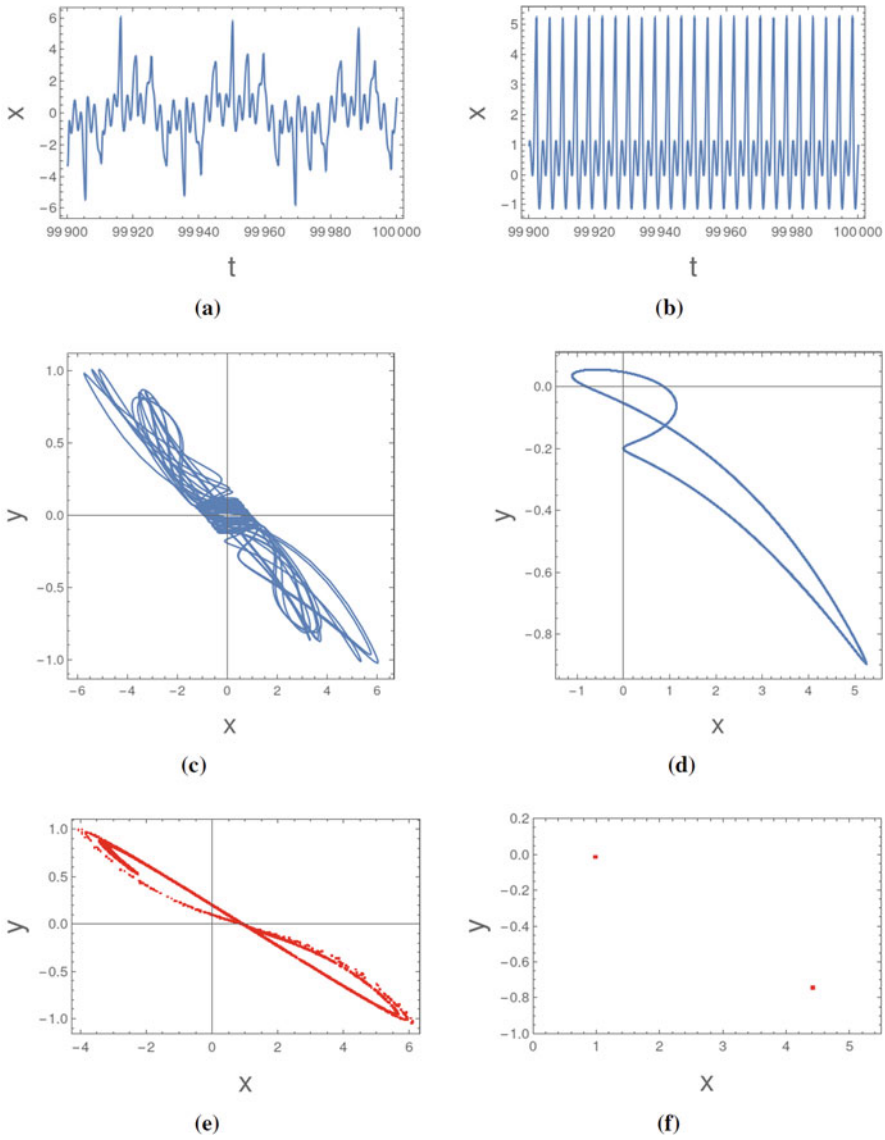


Fig. 11.8 (**a**, **b**) Time responses of x variable, (**c**, **d**) phase portraits, and (**e**, **f**) Poincaré section for $L = 5\,\mathrm{H}$ and $L = 4\,\mathrm{H}$, respectively

bifurcation diagram, as well as from the time responses for $L = 5\,\mathrm{H}$ and $L = 4\,\mathrm{H}$. More specifically, periodic windows inside chaotic regions were observed.

Finally, the last approach was to change the initial conditions x_0, y_0, and z_0. In this approach, it is observed that as the initial conditions are increasing, the chaotic behavior is changing to the regular behavior rapidly. Also, the existence

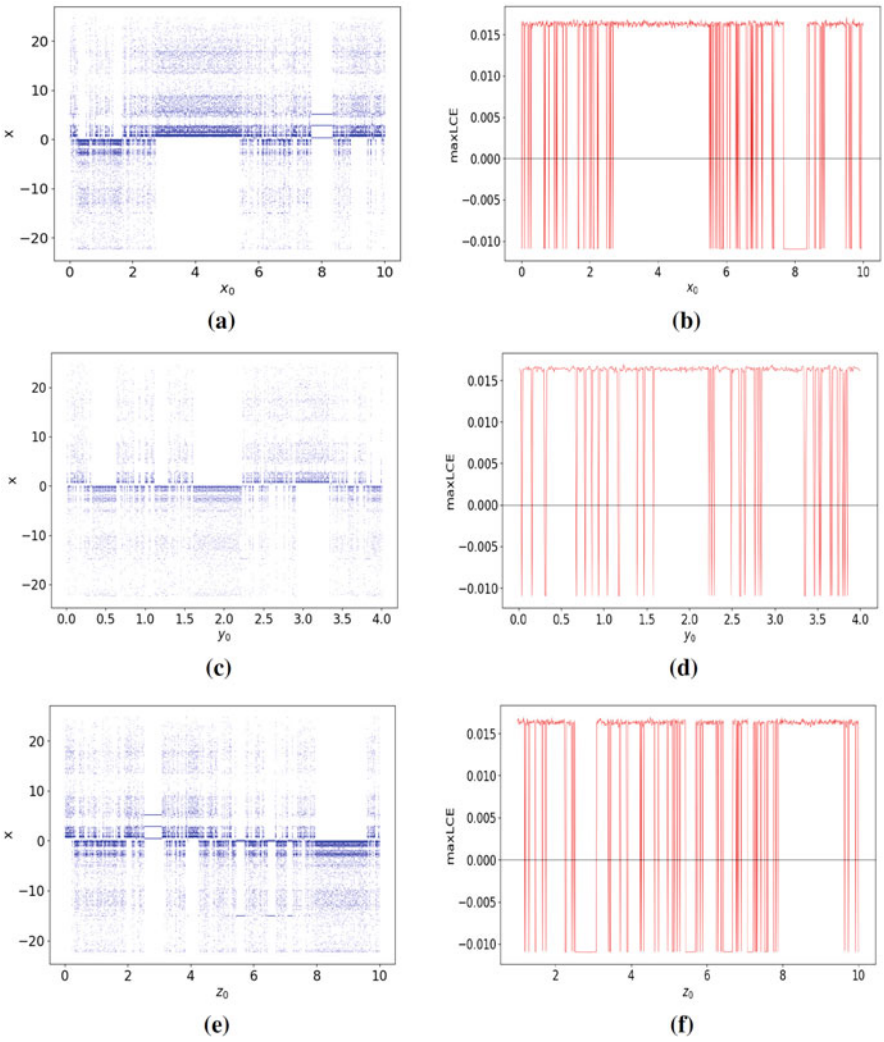


Fig. 11.9 Bifurcation-like diagram and maximal Lyapunov diagram of system in regard to (**a**, **b**) x_0, (**c**, **d**) y_0, and (**e**, **f**) z_0 for $C = 0.01\,\mathrm{F}$

of coexisting attractors was presented. What is more, regions where the behavior is only chaotic and regions where the behavior is only regular were observed. As a recommendation, it would be interesting to investigate the dynamical behavior of the system, by changing the function that describes the nonlinear resistance, as well as by replacing the thermistor with a real memristor.

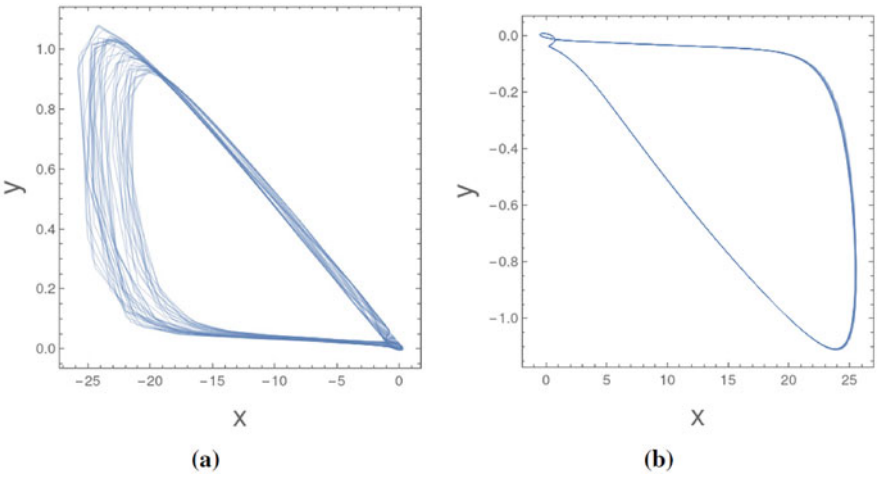


Fig. 11.10 (**a**) Chaotic phase portrait of x–y plane for $x_0 = 0.06$ and (**b**) periodic phase portrait of x-y plane for $x_0 = 8.0$

References

1. L. Chua, Memristor-the missing circuit element. IEEE Trans. Circuit Theory **18**(5), 507–519 (1971)
2. R. Tetzlaff, *Memristors and Memristive Systems* (Springer, Berlin, 2013)
3. S. Parajuli, R.K. Budhathoki, H. Kim, Nonvolatile memory cell based on memristor emulator (2019). Preprint arXiv:1905.04864
4. L.O. Chua, S.M. Kang, Memristive devices and systems. Proc. IEEE **64**(2), 209–223 (1976)
5. J.J. Yang, D.B. Strukov, D.R. Stewart, Memristive devices for computing. Nat. Nanotechnol. **8**(1), 13–24 (2013)
6. J. Kim, Y.V. Pershin, M. Yin, T. Datta, M. Di Ventra, An experimental proof that resistance-switching memory cells are not memristors. Adv. Electron. Mat. **6**(7), 2000010 (2020)
7. L. Minati, L. Gambuzza, W. Thio, J. Sprott, M. Frasca, A chaotic circuit based on a physical memristor. Chaos Solitons Fractals **138**, 109990 (2020)
8. S. Kumar, J.P. Strachan, R.S. Williams, Chaotic dynamics in nanoscale NbO2 mott memristors for analogue computing. Nature **548**(7667), 318–321 (2017)
9. A. Buscarino, L. Fortuna, M. Frasca, L. Valentina Gambuzza, A chaotic circuit based on Hewlett-Packard memristor. Chaos Interdiscip. J. Nonlinear Sci. **22**(2), 023136 (2012)
10. M. Faraday, V. experimental researches in electricity. Philos. Trans. R. Soc. London **122**(122), 125–162 (1832)
11. S. Ruben, Patent US 2,021,491 (1935).
12. J.S. Steinhart, S.R. Hart, Calibration curves for thermistors. Deep-Sea Res. Oceanogr. Abstr. **15**(4), 497–503 (1968). Elsevier
13. M.P. Sah, V. Rajamani, Z.I. Mannan, A. Eroglu, H. Kim, L. Chua, A simple oscillator using memristor, in *Advances in Memristors, Memristive Devices and Systems* (Springer, Berlin, 2017), pp. 19–58
14. J.-M. Ginoux, B. Muthuswamy, R. Meucci, S. Euzzor, A. Di Garbo, K. Ganesan, A physical memristor based Muthusamy–Chua–Ginoux system. Sci. Rep. **10**(1), 1–10 (2020)
15. J. Maaita, I. Kyprianidis, C.K. Volos, E. Meletlidou, The study of a nonlinear duffing–type oscillator driven by two voltage sources. J. Eng. Sci. Technol. Rev. **6**(4), 74–80 (2013)

16. A. Wolf, J.B. Swift, H.L. Swinney, J.A. Vastano, Determining Lyapunov exponents from a time series. Physica D Nonlinear Phenomena **16**(3), 285–317 (1985)
17. Y. Guo, A. Luo, Z. Reyes, A. Reyes, R. Goonesekere, On experimental periodic motions in a duffing oscillatory circuit. J. Vibr. Test. Syst. Dyn. **3**, 55–69 (2019)
18. M. Sandri, Numerical calculation of Lyapunov exponents. Math. J. **6**(3), 78–84 (1996)
19. J.L. Kaplan, J.A. Yorke, Chaotic behavior of multidimensional difference equations, in *Functional Differential Equations and Approximation of Fixed Points* (Springer, Berlin, 1979), pp. 204–227
20. P. Frederickson, J.L. Kaplan, E.D. Yorke, J.A. Yorke, The Lyapunov dimension of strange attractors. J. Diff. Eq. **49**(2), 185–207 (1983)

Chapter 12
Inverse Filter in the Growth of Urban Sprawl with Cellular Automata Model

Eduardo Jiménez-López

Abstract Thanks to the similarity between the population growth dynamics and the operating way of the models based on Cellular Automata (CA), they can be used to simulate processes that involve the growth of the urban area. Changes in land use depend on a set of social and economic factors that combined with CA offer a way to understand a series of very complex challenges, such is the case to determine which is the transition rule that best explains the dynamics of growth of each area of a city, as it is the key to controlling model behavior. A technique called Inverse Filter is proposed, which finds the best transition rule considering criteria such as geometry, position, and amount of population in the space. The evaluation with previous criteria is carried out using the following metrics: Kappa Index, Jaccard Index, Fractal Dimension, and Shannon Entropy, showing the performance of the tool. Urban growth in Toluca city for the period 2003–2017 is considered as a case study using 256 neighborhood rules, generated from a binary count of dimension one.

Keywords Cellular automata · Reverse filter · City growth

12.1 Introduction

Cities are the product of the social pact and their planning must be supported by collective consultation processes [1]. Citizens could hardly contribute actively and informed in the co-design of the city without having automated scientific and technological developments [2]. It is increasingly necessary to have interactive, practical urban models, supported by exact sciences and technological tools, capable of supporting collective consultation and design processes of cities [3, 4].

The traditional models, used to simulate the growth of the urban area, are usually not spatial [5], so they need techniques such as CA, which determine the position

E. Jiménez-López (✉)
El Colegio Mexiquense A.C., Zinacantepec, Estado de México, Mexico

G. Huerta Cuellar et al. (eds.), *Complex Systems and Their Applications*,
https://doi.org/10.1007/978-3-031-02472-6_12

in the nearby neighborhood. The CA construct supported spatial patterns of local interactions between cells. This type of model can handle the growth dynamics of cities through transition rules, which help to explain the influence of neighborhood effects in the study area [6, 7]. It can be said that the transition rules are the fundamental component of the CA models in the growth of the urban area.

CA-based urban growth models are used to understand population growth processes in cities. However, the calibration of these models is a process that requires mathematical techniques that provide objective results. Therefore, the calibration of drivers of urban growth such as accessibility or the calibration of the dynamics of the neighborhood in the models is essential [8].

One reason that CAs are used in various disciplines is because of their effectiveness and precision compared to systems of differential equations, and they also generate more untouchable and intuitive results [9, 10].

The CA dynamically updates the spatial variables involved iteratively over a period. Consequently, the results are not determinant. Each spatial variable has a different impact based on its initial and on neighborhood-level parameters. The CA functions as a spatial allocator by scanning all cells and determines a transition weight for the central cell [11, 12]. The higher the potential of the cells, the more likely it is that the pixels will change to the situation in their interaction with the neighborhood. This mechanism continues until you assign all cells to built or unbuilt areas [4].

The concept of urban expansion is usually associated with dispersed cities, so the literature attempts to identify the factors that influence urban growth toward a certain area [13, 14]. Thus, through the construction of this model it is possible to determine future scenarios that help predict current growth trends and in turn know the consequences of implementing actions to benefit the planning of a city.

Derived from the above, an Inverse Filter is proposed based on the principle of the Mahalanobis distance. The filter works on the principle of finding the best transition rule that simulates the growth of the urban area. To demonstrate the effectiveness of the filter, its advantages are evaluated and presented using the city of Toluca, México, as a case study.

The Inverse Filter is a technique that allows you to compare images as a result of the simulation of scenarios. Such a comparison is made considering a set of variables that measure different characteristics that represent dimensions [15].

When performing the spatialization of the different CA rules, the results are verified using the following metrics: Kappa index, Jaccard index, fractal dimension, and Shannon entropy. These metrics measure characteristics of urban expansion and evaluate changes over time, changes in space, in the geometry, and distribution of the urban area, respectively.

For the development and analysis of this work, a hybrid model is carried out that integrates the AC model to the Geographic Information Systems (GIS), placing the research in the field of Geosimulation [16, 17]. The development allows to simulate the expansion of the urban area in the period 2003–2017 in the metropolitan area of the city of Toluca city.

In addition, the Inverse Filter is primarily intended to facilitate neighborhood influence calibration using an automatic rule detection procedure. Another objective is to develop a practical and useful procedure to automate the processes that require a longer time in the neighborhood calibration process in growth models of the urban area with CA. To demonstrate the efficiency of this approach, it is shown how calibration can help to find the best neighborhood rule in the study city through a procedure that measures different characteristics of the growth of the urban area.

The work is developed under the following structure: Sect. 12.1 highlights some key ideas for this text related to urban models based on Cellular Automata. In Sect. 12.2 the proposed technique called Inverse Filter is explained in detail. The Filter is applied to identify and compare each of the simulations with respect to the observed urban expansion. In Sect. 12.3, the usefulness of the Inverse Filter to explore urban planning scenarios is shown. Finally, in Sect. 12.4 the conclusions of the work are developed, their contributions are highlighted, and the research agenda is presented that allows us to continue advancing toward the availability of urban models that practically support collective consultation and co-design processes of cities from México. Spectral bands are images that capture reflected energy in a different range of the electromagnetic spectrum.

12.2 Material and Method

The study area is the city of Toluca, capital of the State of Mexico located in the center of Mexico. From 2003 to 2017, it increased from 834,399 to 1,338,126 inhabitants [18]. For the development of this research, panchromatic satellite images are used, coming from the Landsat 7 and 8 satellites, composed of a set of spectral bands. Only bands 4, 6, and 7 are used, which highlight urban features when generating RGB color images. The result is the entrance images to the CA [19].

The tool that is proposed and that we call the Inverse Filter has its origins in the local or global comparison stages with the indicators that measure the growth, dispersion, or accuracy in the pixels of the images (i.e., maps) generated by the model that compares the real images. All the indices generated by these indicators are taken, which are grouped and ordered in a matrix; in each row, the indicators of the simulation generated by each rule are placed. These indices are filtered by letting the highest values pass and are placed in the first places of the matrix until the most prominent ones are obtained [20]. We based the study on associating the separation measures between the metrics, turning this multivariate problem into a univariate problem by using a linear combination of measures in vector form involved in the analysis [21].

The filter is coded as a tool and incorporated into the SIG, making use of the PyQGIS package and the Python programming language [19]. With this tool, the computational processing time required to find the best transition rule is reduced. For a CA model, the computational calculation time reached up to three weeks to finding the best rule. The FI model has a calculation time of 8 hours. Therefore

it is mentioned that the calculation time is greatly reduced. The work contributes to the urban planning of the study city, since it normalizes metrics that impact urban growth, in addition to allowing the creation of future applications on GIS environments.

The experimental part of this work applies local Transition Rules, following the logic of a CA, where the neighborhood is defined based on contiguity criteria, that is, the close neighbors have a greater influence on the analysis cell or pixel [22]. In the CA model, a random search of a pixel is carried out within a two-dimensional grid, to which the Transition Rule ∂ is subsequently applied, which is a number in base 10 that goes from the number 0–255, that is, $\partial \in [0, 255]$. This number is translated into its base 2 (binary) equivalent that can be used by the model.

The experimentation of this work uses binary raster images of the city of Toluca in the years 2003–2017. The satellite images obtained with the Landsat 8 satellite are open access; there are some images that are private according to the level of resolution they handle and are delivered by NASA in the reference [23]. The pixels in the image are classified into two categories: 1 (urban land) and 0 (vacant land). For more details on the binarization of maps, see [22]. An example of the initial conditions of the images (maps) in binary format is shown in Fig. 12.1.

The simulation and projection of the urban area in the metropolitan area of Toluca begins with the random search of a pixel and its neighborhood on the raster map where the Transition Rule can be applied. Neighbors are identified: $x - 1$, x, $x + 1$; see Fig. 12.2. The number of processes is determined by the growth trend, the increase in pixels by one, from the 2003 image to the 2017 image. The increase is added to the model and spatialized on the simulation raster map [24].

The adjacency principle and the Transition Rule produce the state of the spatiotemporal structure in each lapse (τ), which in our example is the search for other data in the grid. In a one-dimensional CA, the space consists of a single row of cells to which a basic adjacency principle of two neighbors per cell is applied and to which boundary conditions are also applied.

In the CA, change the value of the next row if there is a neighborhood with a lit cell, as shown in Fig. 12.2. This guarantees that the grid where the binary map

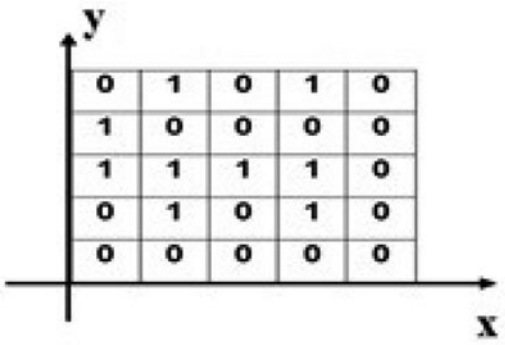


Fig. 12.1 Initial condition of a filtered map for CA analysis

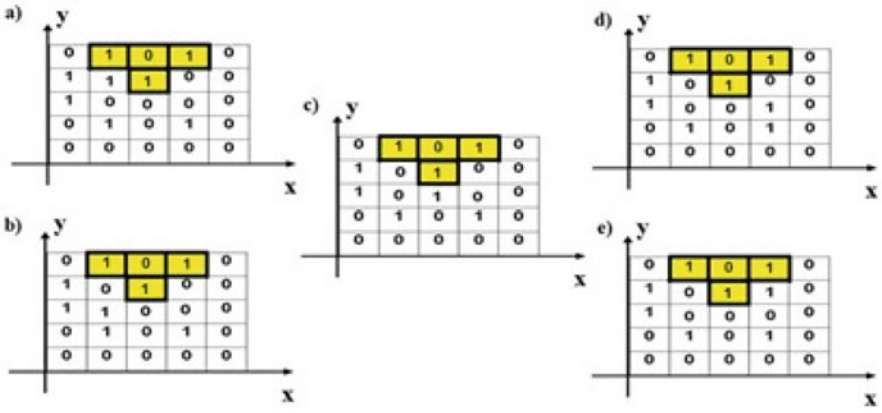


Fig. 12.2 New transition in the CA. Scheduled growth if there are close neighbors

is located does not grow outside the dimensions of the base image (i.e., border problems are overcome).

The Transition Rules used are three bits per binary count. The key is to find the Rule that best simulates the urban expansion process in the study area. To achieve this, at least two conditions are required: i. Test all the Transition Rules (the 256); ii. Use the inverse filter that uses goodness of fit metrics that evaluate the similarity of the results with respect to the observed reality. This allows you to measure the precision of each Transition Rule. The results are used to train the CA and the projection to 14 years can be carried out.

The metrics used in the Reverse Filter are [24]: (1) Cohen's Kappa Index, which measures the similarity between two maps; (2) the Jaccard Index, which measures the equality of the location of the pixels considering their states, in two different times; (3) fractal dimension, which estimates the growth of the urban area; and (4) Shannon entropy, which estimates the distribution of pixels in the urban area image.

12.2.1 Cohen's Kappa Index

Cohen's Kappa Index (k) is a mapped comparison measure that adjusts for the effect of chance. In this work, between binary category maps, the result k can take values between 0 and 1. If the calculated value of the index is close to 1, the degree of agreement between the maps is greater, while if it is close to 0, the disagreement is greater. The result of the Cohen's Kappa index shows the effect of chance and it is possible to know if its results are statistically significant [24, 25].

$$k = \frac{P_0 - P_e}{1 - P_e} \tag{12.1}$$

where P_0 is the observed equality proportion, P_e is the expected equality proportion, and $1 - P_e$ represents the maximum equality or agreement. The numerator of the Cohen's Kappa Index expresses the proportion of observed equality minus the expected one, while the denominator is the difference between total equality and the expected proportion. If this value is equal to 1.0, we would be facing a situation in which the equality between the maps is perfect (100% equality); when the value is 0.0, the maps do not look alike, and if the value is -1.0 a map would be the inverse of the other [25, 26].

12.2.2 Jaccard Index

The Jaccard Similarity Index (I_j) expresses the degree in which two images (e.g., maps) are similar. The range of values for the Jaccard index goes from 0, when the inequality between the maps is total, to 1, when two maps have the same measure of agreement-position. The index is obtained with Eq. 12.2.

$$I_j = \frac{T_{11}}{T_{21} + T_{12} + T_{11}} \tag{12.2}$$

With set theory, the Jaccard Index is easy to analyze. In set A, the objects in its domain are named T_{21}. In set B, the objects in its domain are labeled T_{12}. The objects that are in the union of the two sets are classified as T_{11}. Anything outside of these we label as T_{22}. The Jaccard Index (e.g., Eq. 12.2) calculates two key aspects for comparing maps: the equality of the raster data and its position on the map [24].

12.2.3 Fractal Dimension

Fractals is a study of irregular and fragmented structures that occur at different scales. The appearance of a structure at different scales is called self-similarity, since each of the parts, whatever their degree of approach, presents similarity to the original figure. The degree of irregularity and fragmentation of objects that appear in nature (e.g., in this work maps) can be measured with the fractal dimension (D). Values such as $D = 0$ (point), $D = 1$ (line), $D = 2$ (two-dimensional plane), and $D = 3$ (volume) are well defined in Euclidean space [24].

With the dimension in which one works, it can be considered that the fractal dimension is also a growth dimension, since $D = 1$ is a line and $D = 2$ is a plane totally full of points or a solid plane. Therefore, the fractal dimension that is used is contained between one and two. In another case, if the measure is close to two, we can say that it has many points, it is almost full (i.e., the model projection grew).

The Euclidean dimension of an object relates the unit of measurement used with the geometric value measured $N(L) = (1/L)^D$, where $1/L$ corresponds to the scale, which are the divisions in the plane (i.e., take the image at pixel level). The calculation of D is based on the corresponding measurement of the number of black pixels that cover a certain set (line, surface, or volume), $N(L)$ as a function of the scale. The calculation is only valid in the range in which the relationship between $N(L)$ and $1/L$, which is a potential relationship, is well defined by Eq. 12.3.

$$D = \frac{log(N(L))}{log(1/L)} \tag{12.3}$$

The potential relationship is called the box counting method; for our case, it determines the growth or decrease of the urban area. Fractals are found in many cases in nature, including social systems and socio-spatial structures [27].

12.2.4 Shannon Entropy

Entropy is a concept that has been used to describe the structure and the behavior of different systems [28]. The application of the entropy measure in urban sprawl is proposed to determine the concentration and spatial dispersion of pixels in the image [12, 29, 30].

Shannon entropy measures the maximum possible dispersion in which the binary value of the maps is distributed in the spatial zones. If the index has a value of 1, the value of the variable is evenly distributed among all zones. If it has a value of 0, the variable is concentrated in specific areas. It is expressed with Eq. 12.4.

$$E_n = \frac{\sum_{i=1}^{n} p_i log(1/p_i)}{log(n)} \tag{12.4}$$

where E_n is the relative entropy and p_i is the probability that the variable n is in one of the zones, classes, or categories.

The Shannon entropy calculation is an index that we use remotely with GIS, to measure expansion of the pixels with value 1 in the city map. The value of entropy shows us that there are values in which the efficiency of the city could be compromised [12, 24]. Entropy has values close to 0; the urban area is too concentrated in specific areas, meaning that the city is vulnerable to disasters. On the other hand, if the entropy has a value close to 1, the urban system is well distributed throughout the territory, but it will not be able to adequately allocate the necessary resources for the system to function [28, 31].

12.2.5 Inverse Filter

The inverse filter technique is implemented by means of random neighborhood space sampling, avoiding the over-determinism of the model and producing successful projections according to the applied goodness-of-fit metrics. It improves the calibration of the CA model by identifying the optimal rules that respect criteria from a set of metrics under the same hierarchy level, unlike other studies presented in the literature where they prioritize the first metrics that are estimated, thus eliminating possible potential rules from the first calculations.

Where some of the variables can contain relevant information when related to another (i.e., correlated data). As the case presented by Mahalanobis [32], where the example of a fisherman who wants to measure the similarity between two salmon is shown. With these data, construct a vector (x, y) for each salmon i. The length of the salmon caught is a random variable that takes values between 50 and 100 cm, while its width is between 10 and 20 cm.

If the fisherman uses the Euclidean distance, since the difference in width is less than the length, this measurement would be less important. That is why the fisherman decides to incorporate the data statistics into the distance measure, weighting according to its variance by means of the Mahalanobis distance (i.e., the values are placed inversely to how the main components are placed). Which establishes that the variables with less variance are more important than those with greater variance, trying to equalize the importance of width and length in the final result.

This criterion that seeks to balance the importance of each variable is used in the search for the best transition rule. We have a variety of metrics used to determine which is the best transition rule respect to only one specific criterion (geometry, spatial distribution, pixel change); however, multiple criteria must be considered simultaneously that allow growth to be modeled.

The objective is to determine by statistical calculations the best transition rule that produces the most realistic image of urban sprawl. Thus, it is possible to determine the best vector from all the evaluated rules (i.e., initial data set), which models a scenario in a more similar way with respect to the average vector of the set.

It is established that a rule is optimal when the Filter manages to balance the distance between the centroid and the elements that make up the initial data set, producing symmetric shapes. In Fig. 12.3, we show the graphic representation of how the inverse filter evolves until it finds the optimal rule from four metrics chosen at the beginning of the execution.

In this work four metrics are used, which allow generating 15 possible unrepeatable combinations. However, an indeterminate number of measurements can be incorporated, conditional on using the same type of images and measuring some important aspect of the binary raster image. Depending on the metrics selected by the user, the filter automatically calculates a vector of values for each rule to be evaluated.

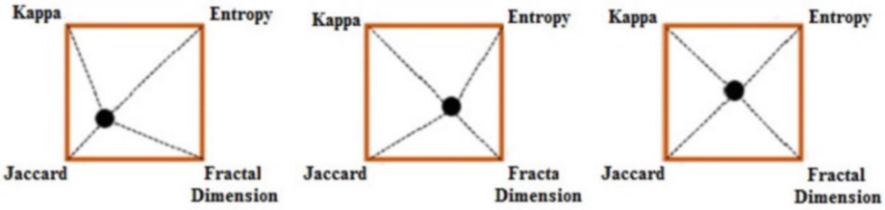


Fig. 12.3 Spatial representation of the search for the best rule from the centroid

Table 12.1 Matrix of fit metrics

Setting value				
	Kappa	Jaccard	Shannon entropy	Fractal dimension
$rule_1$	k_1	J_1	E_1	D_1
$rule_2$	k_2	J_2	E_2	D_2
$rule_3$	k_3	J_3	E_3	D_3
...	...	...	...	...
...	...	...	...	...
$rule_{256}$	k_{256}	J_{256}	E_{256}	D_{256}

The values of each metric are calculated based on the comparison image with respect to the binary rules, in such a way that, at the end of the calculations, a matrix of metrics is obtained, as shown in Table 12.1.

Once the goodness of fit metric matrix has been obtained, the centroid C must be calculated (i.e., the value of the metrics that approach equilibrium, or if they fit into a geometric figure in the center), which is obtained from the average of each column x_i, of matrix A, thus generating a vector of average metrics.

The way to calculate the Mahalanobis distance within the inverse filter for two random variables with the same probability distribution $\overrightarrow{x}$ and $\overrightarrow{y}$, with a covariance matrix Γ is defined by Eq. 12.5.

$$d_m(\overrightarrow{x}, \overrightarrow{y}) = \sqrt{(\overrightarrow{x} - \overrightarrow{y})^T \Gamma^{-1} (\overrightarrow{x} - \overrightarrow{y})} \tag{12.5}$$

The process is to calculate the Mahalanobis distance (d_M), between the centroid C and the vectors that make up the metric matrix A, using Eq. 12.6.

$$d_M = \sqrt{S(\Gamma^{-1} \cdot X)} \tag{12.6}$$

where S is the diagonal of Γ, which is a matrix of the form $\Gamma = \dfrac{X}{n-1}$ and is calculated from the form $X = B \cdot B^T$, while B is determined from $B = A - C$. Once the distance vector has been obtained, the smallest value of the set must be selected, which corresponds to the rule n of the matrix A closest to the centroid, this

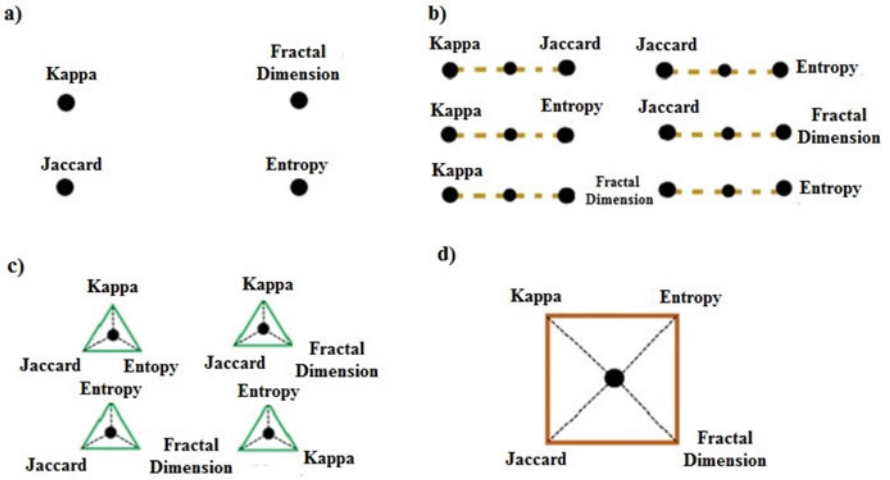


Fig. 12.4 Graphic representation of the possible combinations that can be generated with the tool, combinations with four metrics (**a**)–(**d**)

being the best vector of the set. If more than one equal minimum value is found, all matches are considered as optimal rules.

It should be mentioned that the dimension of matrix A depends on the number of metrics chosen at the beginning of the execution of the tool, corresponding to the number of columns j of matrix A. In Fig. 12.5, the graphic representation of the possible combinations that can be generated from the four metrics used in this work is shown (Fig. 12.4).

12.3 Results

The contribution of a technique that allows to identify the transition rule that best replicates the observed reality is highlighted. This filtering model incorporates global and local indicators that identify the coincidence of pixels, which allows showing the transition rule that reports the best fit between the observed expansion of the city and that generated by CA.

There are growth restrictions in the city of Toluca that remain constant for fourteen years, and they are shown in black in Fig. 12.5. Restricted zones are added to CA that spatialize the expansion. In these restrictions on the land they are not perceptible in the printed medium, areas in which it cannot be built, so they give some examples. A restricted zone is near the center of Toluca, which is the Cerro de la Teresona area and is part of a national park, shown in Fig. 12.5 as a circle

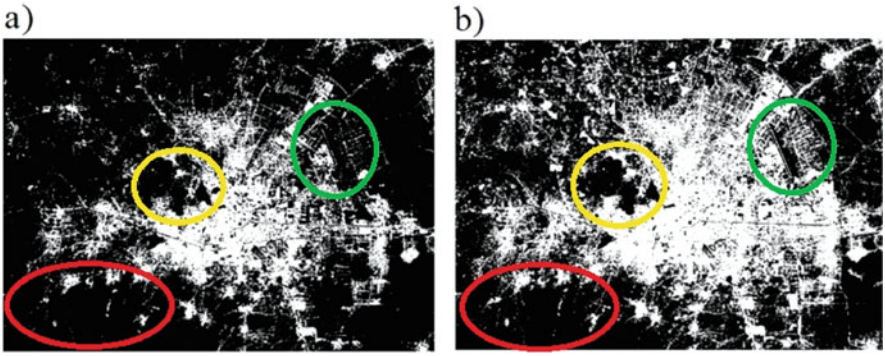


Fig. 12.5 Maps of the city of Toluca 2003 (**a**) and 2017 (**b**)

in yellow color. The slope of this area makes urbanization very complicated, in addition to being a park in the metropolitan area. Another restricted area is the Nevado de Toluca volcano, which is in the lower-left part of the image, shown in Fig. 12.5 as a circle in red color. This area is a national park and construction is not allowed there. A third zone restricted to urbanization is the one located in the Lerma area, to the east of the city, since it has land destined for the expansion of the Toluca International Airport, shown in Fig. 12.5 as a circle in green color.

With the use of the Inverse Filter tool, the comparison, growth, and distribution indicators are estimated, which validates the growth rule that best adjusts to real growth, and this shows how effective the methodology is in expanding the urban patch. Being the best rule for the year 2017, 109. Which has a vector of goodness of fit metrics at a distance from the centroid with an average value of 0.71, being the closest of all those that make up the initial data set.

To perform a test bench of the model, it is necessary to make a projection using the map of Toluca in 2003 as a reference to sketch the map in 2017 and thus make the comparison with the real map of 2017. The simulation of the growth of the urban area for 2017 with the best rule is found by the Inverse Filter, which is a contribution to the model. To perform a diffusion or expansion, the values calculated in a growth trend in the image pixels are taken, and only the values that change from zero to one are considered, 596,261 pixels and those that remained in one, 624,371 pixels: a total of 1,220,632 pixels that will change on the 2003 map.

With the projection of the city of Toluca for 2017, to perform with the best rule found by the Filter, we can see how efficient the pixel distribution method is. The best values are the following, the Kappa value is equal to 0.65, Jaccard 0.75, Shannon entropy 0.92, and fractal dimension 1.83, with a standard deviation of 0.0591 that measures the variation with respect to the average of the indicators used.

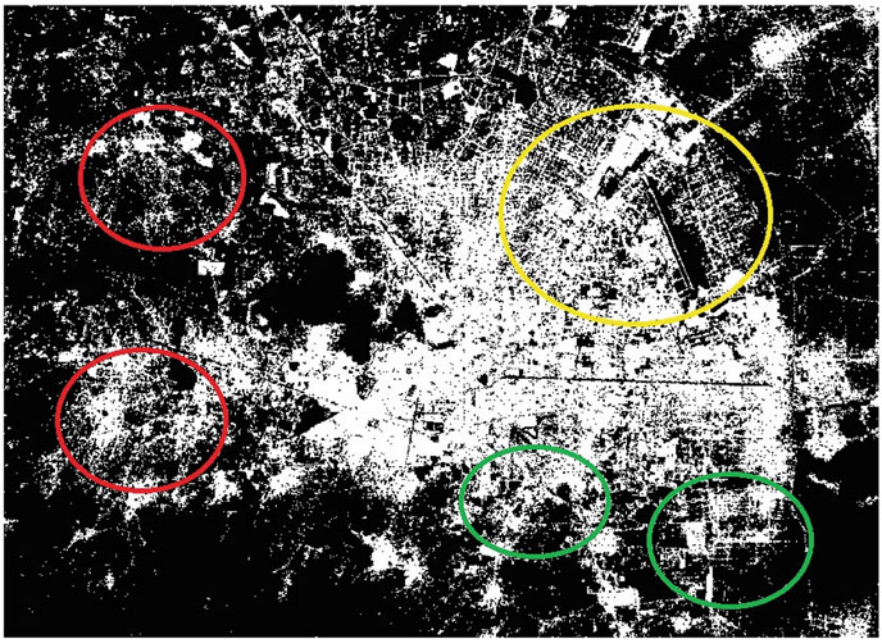


Fig. 12.6 Toluca projection for 2031

The values of the comparison indicators are good according to the scale shown in [24], and the rule allows us to distribute the pixels with less uncertainty in the expansion of the urban area for Toluca in a period of fourteen years. An important aspect that should be highlighted is the calculation of the fractal dimension to determine how much the urban area grew, this tool has been widely used to measure rough geometric figures or with breaks in lines, and in this work it is used to determine growth that is an important aspect in the branch of science where it works with the growth of the urban area. The growth measured with the fractal dimension with the 2003 satellite image is 1.74 and for the 2017 satellite image it is 1.85.

The proposed Inverse Filter model for AC is efficient and has very good results. We can generate a projection of the city of Toluca for 2031, Fig. 12.6, with the best rule found, which is 109. This image shows an equilibrium in the values that changed from one to zero and vice versa; therefore, it can be said that in a period of 14 years the growth rate of the city of Toluca continues in an upward line and is verified with the fractal dimension showing in value for the projection of 1901.

In the simulation carried out for 2031 it covers 54,262 hectares. An increase of 24,275 hectares, which means that in 14 years the city is expected to grow almost 81% compared to 2003. The growth of the previous 14 years (from 2003 to 2017) had a similar growth rate. It must be remembered that the model works with a trend growth; therefore, it is important to know where the city will grow. The simulation of the model implies a more accelerated expansion in the city of Toluca in its number of occupied hectares.

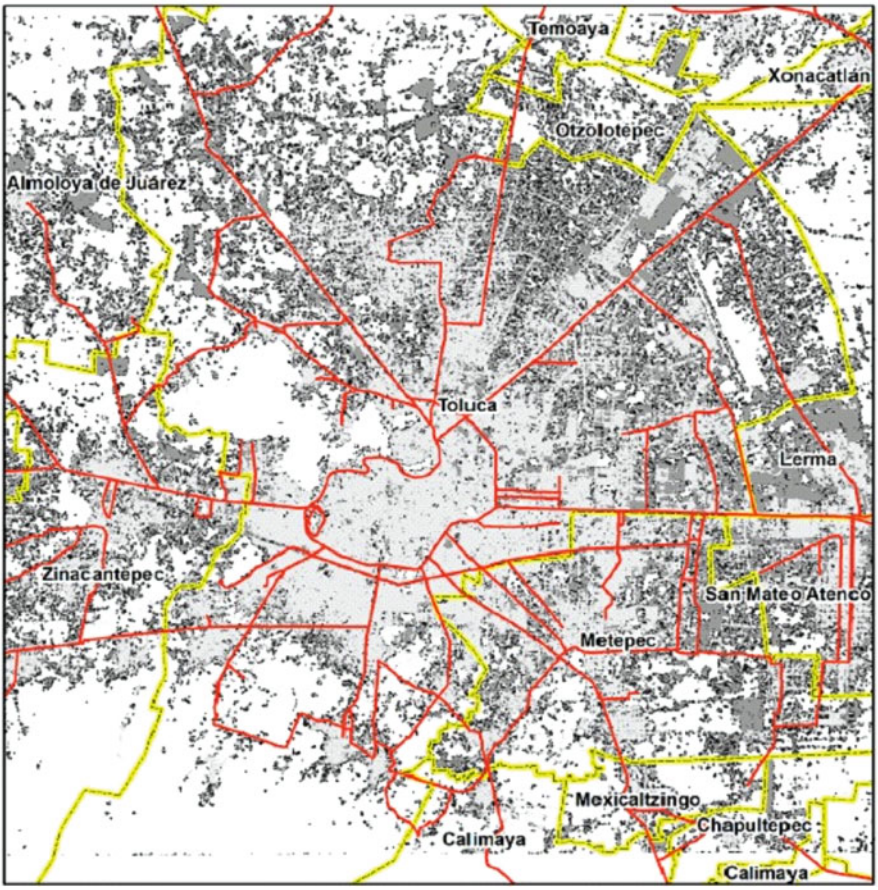


Fig. 12.7 City of Toluca: visual comparison map in the growth period 2003–2031, use of the Cellular Automata model

Figure 12.6 shows areas enclosed by colored circles that highlight five peri-urban areas that are experiencing expansion. One in the northwestern periphery of the city (in Almoloya de Juárez: space rich in agricultural land), and another in the west (in Zinacantepec), circles in red. One more in the extreme east (in Lerma), circle in yellow color and finally two in the southeast of the urban area (Metepec, Mexicaltzingo, and Calimaya: areas rich in agricultural lands), circles in green color.

In Fig. 12.7 the growth of the urban area can be seen with the naked eye. It responds well to the basic pattern generated by the best rule found by the filter. The urban area maps of the city of Toluca in 2003 and 2017 are superimposed giving it two shades of gray on the map in Fig. 12.7, and it can be said that it is an area built in 14 years, captured by the satellite photo. The new urban areas generated by AC and the best rule found can be seen on the map are the pixels in black, the product of the simulation.

With a visual inspection, we can realize that there are three areas that maintain their appearance almost unchanged until 2031: the center of the city of Toluca is an area that has been maintained in 28 years and is the Cerro de la Teresona; due to its elevations, it is difficult to build there and there are other restrictions. In the second area where accelerated growth is not seen, it is the Nevado de Toluca volcano, which is in the lower-left part of the image, it cannot be built. In the Lerma area, which has land destined for the Toluca airport, the same where construction is prevented (Fig. 12.7, seen from the front on the right side almost in the center).

Comparing the images of urban growth maps is very misleading to the observer, but an effort is made to jump this obstacle in the program that was developed, as shown in Fig. 12.7. The visual and numerical comparison showed how the CA model made with the Inverse Filter used in the growth of the urban area responds to the basic pattern that occurs in reality, but showing a larger and more compact growth. In addition, with this model, the new growth of the urban area adheres to existing constructions and new isolated urbanizations disappear due to lack of neighborhood or due to space restrictions to build.

12.4 Conclusions

In this paper, an innovative technique was adopted to explore the growth of the urban area in the city of Toluca: Inverse Filter in Cellular Automata. The objective was to show how this growth-projection technique works, taking as an example the Metropolitan Area of Toluca, which can be said to have a very high growth rate because it is a millionaire city in relation to its inhabitants (i.e., more than one million inhabitants in the city)[25]. The results show that the analysis approach with Inverse Filter offers valuable information and an alternative vision to the traditional approach of growth with regions, and it focuses mainly on the growth of the urban area with pixels in binary maps based on rasters.

In CA with the Inverse Filter, the most remarkable thing is its spatio-temporal analysis, which is very difficult to find in techniques that project maps or that carry out city growth. The search for trends in the projections of the occupied pixels leads us to an analysis over time, and it is what is used to only add the space with the best rule found by the Inverse Filter. Projections are generated over 14 years, from the available city maps.

The union of a geographic information system with CA together with the Inverse Filter leads us by the hand to generate a software tool programmed in Python. This software automates the analysis processes and generates an interesting tool that can be used in different areas of knowledge.

The assumption that we compare maps is correct, since all the work its main ayasgo that mathematical techniques can be applied to determine the growth of the urban sprawl a branch of geography, where a study of this type is little used. Furthermore, what is shown in Fig. 12.7, where a visual comparison between maps is made, shows the dynamics of change of the urban area over 28 years and the

areas that are restricted in the same period. The comparison of the maps and the generation of indices show how efficient the model is; however, the real variable in which you cannot have control is in obtaining the maps for the analysis, and it all depends on whether you have the same shooting conditions in the shot, but this cannot be easily achieved.

Map comparison procedures led to the exploration of techniques little used in cartographic research: the fractal dimension and Shannon entropy, whose use has been an important contribution to show the growth diffusion of the urban area.

When finding the best rule, which is 109, it is considered the best due to the comparison values that are generated by the aforementioned indicators. Es decir, los valores de comparación tanto globales como locales pueden tener la misma distancia al centroide de la estructura geométrica lograda con los cuatro indicadores. The rule is a neighborhood characteristic of the city under study, it determines how the neighborhood is distributed in the city. In later works, the implications of finding this rule for the city of Toluca will be determined in more detail, determining if the city is self-similar, fractal, and all fractional measurements obtained.

It can be concluded that the combination of mathematical tools and GIS in the analysis, supervision, and control of urban phenomena constitutes a tool of great value, especially if the necessary resources are available. If maps or images of good resolution, both spatial and temporal, are available, very precise information can be extracted and processed very reliably through the use of GIS. These data, the greater their precision, they greatly facilitate the obtaining of indices numbers and show a correct projection of the cities under study.

References

1. M. Wagner, W.T. de Vries, Comparative review of methods supporting decision-making in urban development and land management. Land. **8**(8), 123 (2019)
2. N. Unsworth, K. Fukuda, E. Awh, E.K. Vogel, Working memory and fluid intelligence: Capacity, attention control, and secondary memory retrieval. Cogn. Psychol. **71**, 1–26 (2014)
3. S. Angilella, S. Corrente, S. Greco, R. Słowiński, Robust ordinal regression and stochastic multiobjective acceptability analysis in multiple criteria hierarchy process for the Choquet integral preference model. Omega. **63**, 154–169 (2016)
4. E. Jiménez, C. Garrocho, T. Chávez, Autómatas Celulares en Cascada para modelar la expansión urbana con áreas restringidas. Estudios demográficos y urbanos **36**(3), 779–824 (2021)
5. M.M. Aburas, Y.M. Ho, M.F. Ramli, Z.H. Ash'aari, The simulation and prediction of spatio-temporal urban growth trends using cellular automata models. Rev. Int. J. Appl. Earth Obs. Geoinf. **52**, 380–389 (2016)
6. E. Besussi, A. Cecchini, E. Rinaldi, The diffused city of the Italian north-east: identification of urban dynamics using cellular automata urban models. Comput. Environ. Urban Syst. **22**(5), 497–523 (1998)
7. T. Xu, J. Gao, Directional multi-scale analysis and simulation of urban expansion in Auckland, New Zealand using logistic cellular automata. Comput. Environ. Urban Syst. **78**, 101390 (2019)

8. M.S. Roodposhti, R J. Hewitt, B.A. Bryan, Towards automatic calibration of neighbourhood influence in cellular automata land-use models. Comput. Environ. Urban Syst. **79**, 101416 (2020)
9. H. Shafizadeh-Moghadam, A. Asghari, A. Tayyebi, M. Taleai, Coupling machine learning, tree-based and statistical models with cellular automata to simulate urban growth. Comput. Environ. Urban Syst. **64**, 297–308 (2017)
10. B. Rimal, L. Zhang, H. Keshtkar, B.N. Haack, S. Rijal, P. Zhang, Land use/land cover dynamics and modeling of urban land expansion by the integration of cellular automata and Markov chain. ISPRS Int. J. Geo-Inf. **7**(4), 154 (2018)
11. C.P. Newland, H.R. Maier, A.C. Zecchin, J.P. Newman, H. van Delden, Multi-objective optimisation framework for calibration of cellular automata land-use models. Environ. Modell. Softw. **100**, 175–200 (2018)
12. A. Mohamed, H. Worku, Simulating Urban land use and cover dynamics using cellular automata and Markov chain approach in Addis Ababa and the surrounding. Urban Climate. **31**, 100545 (2020)
13. F. Aguilera, Predicción del crecimiento urbano mediante sistemas de información geográfica y modelos basados en autómatas celulares. GeoFocus. **1**(6), 81–112 (2006)
14. C. Martner, Expansión dispersa, ciudad difusa y transporte: el caso de Querétaro, México. EURE. **42**(125), 31–60 (2016)
15. R. De Maesschalck, D. Jouan-Rimbaud, D.L. Massart, The Mahalanobis distance. Chemom. Intell. Lab. Syst. **50**(1), 1–18 (2000)
16. I. Benenson, P. Torrens, *Geosimulation: Automata-Based Modeling of Urban Phenomena* (Wiley, Hoboken, 2004)
17. M. Batty, *Cities and Complexity: Understanding Cities with Cellular Automata, Agent-Based Models, and Fractals* (The MIT Press, Cambridge, 2007)
18. INEGI, Dirección Regional Centro Sur / Coordinación Estatal México. Comunicado de Prensa Nùm. 55/21, 26 de Enero de 2021, Toluca, Estado de México (2021)
19. E. Jardón, E. Jiménez, M. Romero, Spatial Markov chains implemented in GIS, in *International Conference on Computational Science and Computational Intelligence (CSCI)* (IEEE, Piscataway, 2018), pp. 361–367
20. Ó. Martín-Rodríguez, J. C. Fernández-Molina, M.Á. Montero-Alonso, F. González-Gómez, The main components of satisfaction with e-learning. Technol. Pedagog. Educ. **24**(2), 267–277 (2015)
21. R.A. Leiva, M. Herrera, Generalización de la distancia de Mahalanobis para el análisis discriminante lineal en poblaciones con matrices de covarianza desiguales. Revista de la Sociedad Argentina de Estadística. **3**, 1–24 (1999)
22. S. Wolfram, Statistical mechanics of cellular automata. Rev. Modern Phys. **55**(3), 601 (1983)
23. EOS – Earth Observing System (2018). https://eos.com/landviewer/?s=Landsat8. Access 7 Jul 2018
24. E. Jiménez López, Cadenas de Markov espaciales para simular el crecimiento del Área Metropolitana de Toluca, 2017–2031. Economía, sociedad y territorio **19**(60), 109–140 (2019)
25. E. Jiménez, C. Garrocho, T. Chávez, Modelando la expansión urbana con autómatas celulares: aplicación de la Estación de Inteligencia Territorial (CHRISTALLER®). Geografía y Sistemas de Información Geográfica (GeoSig) **12**(10), 1–26 (2018)
26. M. Benchoufi, E. Matzner-Lober, N. Molinari, A.S. Jannot, P. Soyer, Interobserver agreement issues in radiology. Diagn. Interv. Imaging **101**(10), 639–641 (2020)
27. M.F. Goodchild, D.M. Mark, The fractal nature of geographic phenomena. Ann. Assoc. Amer. Geogr. **77**(2), 265–278 (1987)
28. M. Batty, Building a science of cities. Cities **29**, S9-S16 (2012)
29. H.S. Sudhira, T.V. Ramachandra, K.S. Jagadish, Urban sprawl: metrics, dynamics and modelling using GIS. Int. J. Appl. Earth Obs. Geoinf. **5**(1), 29–39 (2004)
30. H. Sun, W. Forsythe, N. Waters, Modeling Urban land use change and Urban sprawl: Calgary, Alberta, Canada. Netw. Spatial Econ. **7**(4), 353–376 (2007)

31. P. Cabral, G. Augusto, M. Tewolde, Y. Araya, Entropy in urban systems. Entropy. **15**(12), 5223–5236 (2013)
32. P.C. Mahalanobis, On the generalized distance in statistics. National Institute of Science of India (1936)

Chapter 13
A Metapopulation Network Model with Seasonal Succession to Analyze Dengue Disease in México

Andrés Anzo Hernández

Abstract Most epidemics occur in geospatial structured populations where human mobility plays a crucial role in the spread of the disease. One methodology to analyze the dynamics of the spread in these systems is with the use of metapopulation network models, where the geographic territory is partitioned into patches, each one inhabited by individuals and mosquitoes; next one considers a network of patches by adding links among them if humans of a given patch move toward another patch. On the other hand, dengue fever is a mosquito-borne disease that prevails in tropical and subtropical regions, mainly in urban or rural environments with social or economic poverty conditions. Furthermore, empirical data indicate a relation among climate variables and mosquito population growth, making dengue outbreaks a seasonal succession dynamic system. In this work a metapopulation network model is proposed where the dengue disease in each patch is represented with a SIR-SI epidemic model. In order to consider the seasonality feature of a dengue disease, a piecewise constant signal is introduced in order to switch the growth dynamics of mosquito population into two intervals called favorable and unfavorable reproduction periods. A perspective of how, with climate data and other geographical information that benefit mosquitoes live cycle, could be incorporated in the model in order to propose controlling strategies against dengue transmission in México.

Keywords Complex networks · Mathematical epidemiology · Dengue disease

A. A. Hernández (✉)
Investigadoras e Investigadores por México, CONACyT-BUAP, Facultad de Ciencias Físico-Matemáticas, Centro de Modelación Matemática y Computacional (FCFM-CEMMAC), Puebla, Mexico

G. Huerta Cuellar et al. (eds.), *Complex Systems and Their Applications*,
https://doi.org/10.1007/978-3-031-02472-6_13

13.1 Introduction

Dengue (DEN), Zika (ZIKV), and Chikungunya (CHIKV) are the main three arboviruses that are transmitted to humans via the bite of two mosquito species named *Aedes aegypti* (Stegomyia aegypti) and *Aedes albopictus* (Stegomyia albopicta). Nowadays, DEN is one of the major mosquito-transmitted diseases with a significant international public health concern; an estimate of 125 tropical and subtropical countries are affected and 100 million symptomatic infections per year are reported [1, 2]. In particular, México in 2019 have had 30 thousand 609 confirmed cases of DEN, where Jalisco, Veracruz, and Quintana Roo were the most affected states [3].

Urbanization, population growth, and human mobility are important factors that contribute to the increase in DEN outbreaks. In this context, the metapopulation network models, widely used in Ecology, have been considered in the study of disease outbreaks [4, 5]. In order to formulate a geospatial metapopulation network model, firstly the geographic territory is partitioned into patches, each one occupied by individuals; next one considers a network of patches by adding links among them if humans of a given patch move toward another patch. One of the most used mobility models is the Lagrangian approach, where people spend a fraction of their day at some patch (by work or personal reasons) and then return to its own patch after that [6, 7]. That is, human mobility is described by the dwell-time parameters p_{ij} that stand for the fraction of the day that residents from patch i spend in patch j.

On the other hand, recent studies indicate that *Aedes aegypti* life cycle is remarkably sensitive to climate state, such as rainfall, temperature, humidity, photoperiod among others [8]. That is, environmental conditions can trigger a change in the mosquito population growth, and as a consequence, make DEN a seasonality viral infection. In this sense, winter season represents an unfavorable period for mosquito reproduction, and, otherwise, summer season switches the population growth dynamics by increasing the number of mosquitos. It is worth remarking here that *Aedes aegypti* has an adaptable mechanism called diapause, which lets the population to survive those unfavorable climate conditions. This mechanism is related to the dynamic state of low metabolic activity of mosquito eggs [9]. In this work a geospatial metapopulation network based on a compartment SIR model for human and SI for vectors is proposed, where a switching signal is introduced in the model in order to trigger a change in the dynamics of mosquito population growth. That is, a hybrid switched system for the SIR-SI model in a metapopulation network is proposed. The model includes a favorable season period where the mosquito population grows according to a logistic model, and an unfavorable season where the population behavior of mosquitos switches to an exponential decrease dynamics. The model is analyzed for the case of two connected patches with equal and different values of the carrying capacity parameter.

This paper is organized as follows: In Sect. 13.2 the proposed model is presented. In Sect. 13.3 the case of two connected patches is analyzed. Section 13.4 presents discussions about the role of climate, diapause, and perspectives. In Sect. 13.5 some concluding remarks are given.

13.2 Model Formulation

Let Ω be a geospatial region where DEN disease could be transmitted. The region could be any rural or urban environment, municipality, city, or a country that can be divided into N disjoint subregions $\Omega = \cup_{i=1}^{N} \Omega_i$ called patches. The connection between a pair of patches represents the human mobility among such regions, in such a way that we can construct a network of geospatial regions as the one shown in Fig. 13.1, where a schematic representation of the human mobility between patches in the state of Guerrero, México is drawn. In what follows a mathematical model for Dengue disease in a single patch is described, and next, the network of patches with human mobility is traced.

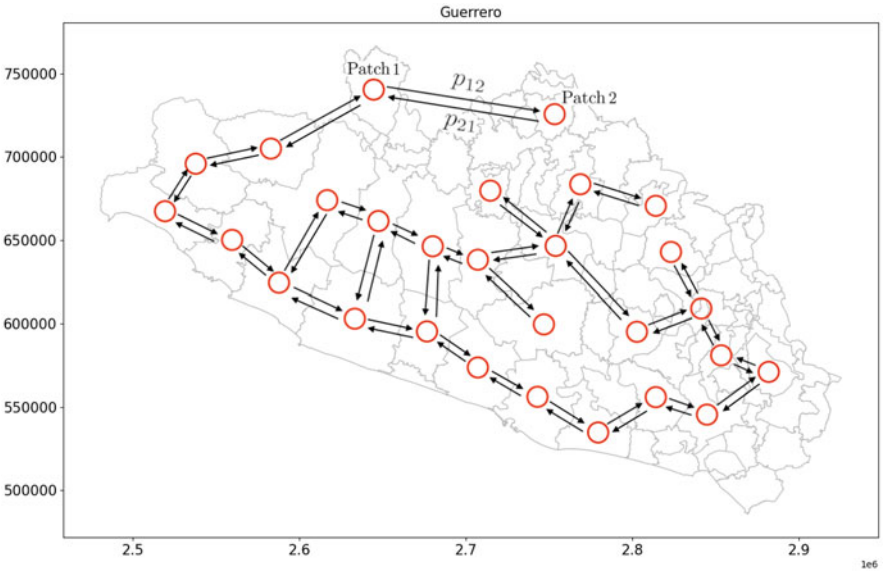


Fig. 13.1 A schematic representation of the human mobility between patches in the state of Guerrero, México

13.2.1 A Single Population Model

Assume that for a given sub-geographical region Ω_i, with $i \in \{1, 2, \ldots, N\}$, humans and vectors are homogeneously mixed. Furthermore, consider that human population in patch i can be classified at any time t into susceptible $S_i(t)$, infectious $I_i(t)$, and recovered $R_i(t)$ individuals; while mosquito population is classified into susceptible $M_i(t)$ and infectious $V_i(t)$. Thus, we have $N_{hi}(t) = S_i(t) + I_i(t) + R_i(t) = cte$; and $N_{vi}(t) = M_i(t) + V_i(t) = cte, \forall t$.

In this context, the human population dynamics is described by using an SIR model and mosquito dynamics by using an SI model. Then, the coupling of both models gives rise to the following system of differential equations:

$$\text{humans} \begin{cases} \dot{S}_i = \mu_{hi}(N_{hi} - S_i) - \beta_i V_i \dfrac{S_i}{N_{hi}}, \\ \dot{I}_i = -\gamma_i I_i - \mu_{hi} I_i + \beta_i V_i \dfrac{S_i}{N_{hi}}, \\ \dot{R}_i = \gamma_i I_i - \mu_{hi} R_i, \end{cases} \tag{13.1}$$

$$\text{vectors} \begin{cases} \dot{M}_i = \Lambda_i - \mu_{vi} M_i - \beta_i M_i \dfrac{I_i}{N_{hi}}, \\ \dot{V}_i = -\mu_{vi} V_i + \beta_i M_i \dfrac{I}{N_{hi}}; \end{cases} \tag{13.2}$$

where, β_i is the mosquitoes effective vector biting rate in patch i; μ_h and μ_v are the per-capita birth and natural mortality rates in humans and vectors respectively, γ is the per-capita human recovery rate; and Λ_i is the constant recruitment rate that represents the natural growth of susceptible vectors. Then, $\beta V_i S_i / N_{hi}$ is the total number of susceptible hosts that become ill by the bite of the infected vectors, and $\beta M_i I_i / N_{hi}$ is the total number of susceptible vectors that contract the arbovirus due to their interaction with the fraction I_i / N_{hi} of infected humans. In Table 13.1 some numerical values from the above described parameters reported in the literature for the Dengue case is presented [10] and, in Fig. 13.2, the numerical solution of the SIR-SI model (13.1)–(13.2), by using the fourth order Runge−Kutta method (RK4) and the parameters values given in Table 13.1, with $\beta_i = 0.67$, and initial condition $S_i(t = 0) = 100{,}000$, $I_i(t = 0) = 1$, $R_i(t = 0) = 0$, $M_i(t = 0) = 80{,}000$, $V_i(t = 0) = 0$ is shown.

Table 13.1 Description and parameters values of the SIR-SI model (Eqs. (13.1)–(13.2)) for the Dengue case [10]

Parameter	Description	Numerical value
μ_{hi}	Per-capita birth/mortality rate of humans (per day, per capita)	4.57×10^{-5} humans/days
μ_{vi}	Per-capita mortality rate of adult female mosquitoes (per day, per capita)	1/8 mosquitos/days
γ_i	Humans recovery rate (per day, per capita)	1/7 humans/days
Λ_i	The constant recruitment rate of vectors	1000 mosquitoes/days
β_i	Effective vector biting rate (global number of bites, per day, per mosquito)	[0.2, 0.67] bites/mosquitoes×days

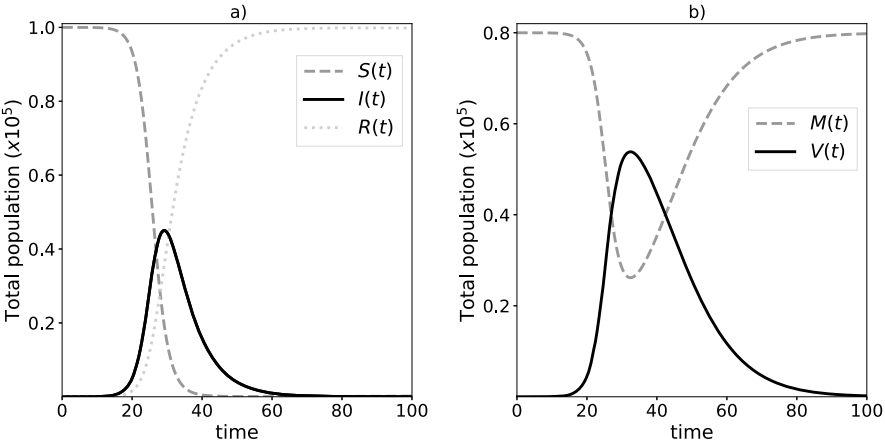


Fig. 13.2 Numerical solution of the SIR-SI model (13.1)–(13.2) with the fourth order Runge–Kutta method (RK4) and with the parameters values given in Table 13.1; with $\beta = 0.67$, and initial condition $S_i(t = 0) = 100{,}000$, $I_i(t = 0) = 1$, $R_i(t = 0) = 0$, $M_i(t = 0) = 80{,}000$, $V_i(t = 0) = 0$. (**a**) Humans. (**b**) Mosquitoes

13.2.2 A Metapopulation Network Model

A network composed of N patches is considered, where each patch is inhabited by a homogeneously mixing population of N_{hi} humans and N_{vi} mosquitoes, where $i \in \{1, 2, \ldots, N\}$ is the patch label. As before, $S_i(t)$, $I_i(t)$, and $R_i(t)$ are the number of humans residents in patch i that are susceptible, infected, and recovered and $M_i(t)$ and $V_i(t)$ the number of susceptible and infected mosquitoes in the i-th patch at time t, respectively.

It is assumed that there exists a flux of human mobility among patches, while vectors remain in the same patch all the time. In order to incorporate human mobility in the SIR-SI model (13.1)–(13.2), the dwell-time matrix $\mathbf{P} = [p_{ij}]_{N\times N}$ is introduced, whose entries $p_{ij} \geq 0$, for $i, j = 1, \ldots, N$, describe the fraction of the

day that hosts in patch i spend in patch j [4]. The dwell-time matrix should satisfy $\sum_{j=1}^{N} p_{ij} = 1$ for all i.

The effect of human mobility in the spread of a vector-borne disease is introduced by substituting the last term at right hand of the Eqs. (13.1) by the function f_i, which describes the human mobility over the metapopulation network of susceptible residents from i-th patch, and how they contract the arbovirus due to the bite of the infected vectors that live on the patches that they visit. Additionally, the last term at right hand of the Eqs. (13.2) is replaced by the function g_i, which describes the human mobility over the metapopulation network of the infected residents from any patch, and how they interact with susceptible vectors in patch i. In this context, the SIR-SI model with human mobility among the N patches is expressed as the following system of $5N$ non-linear ordinary differential equations:

$$\text{Patch } k \begin{cases} \dot{S}_k &= \mu_{hk}(N_{hk} - S_k) - f_k(S_k, \mathbf{I_v}), \\ \dot{I}_k &= -(\gamma_k + \mu_h) I_k \quad + f_k(S_k, \mathbf{I_v}), \\ \dot{R}_k &= \gamma_k I_k - \mu_{hk} R_k, \\ \dot{M}_k &= \Lambda_k - \mu_{vk} M_k \quad - g_k(M_k, \mathbf{I_h}), \\ \dot{V}_k &= -\mu_{vk} V_k \qquad\quad + g_k(M_k, \mathbf{I_h}); \end{cases} \tag{13.3}$$

where $k = 1, 2, \ldots, N$; and μ_{hk}, μ_{vk}, and γ_k are the parameters described in the Table 13.1 for the k-th patch; and $\mathbf{I_v} = (V_1, V_2, \ldots, V_N)$ and $\mathbf{I_h} = (I_1, I_2, \ldots, I_N)$.

On the other hand, let $w_{hi} = \sum_{\tau=1}^{N} p_{\tau i} N_{h\tau}$ be the effective number of humans that spend their time in patch i, including both own residents and visitors of the neighboring patches. That is, w_{hi} indicates the amount of humans that daily visit patch i. Then, the term $S_k p_{kj}/w_{hj}$ is the proportion of susceptible humans that travel from patch k to patch j and, $I_j p_{jk}/w_{hk}$ is the proportion of infected humans of patch j that travel to patch k. Then the functions f_k and g_k for Eqs. (13.3) are written as

$$f_k(S_k, \mathbf{I_v}) = \sum_{j=1}^{N} \beta_j V_j \frac{S_k p_{kj}}{w_{hj}}, \tag{13.4}$$

and

$$g_k(M_k, \mathbf{I_h}) = \sum_{j=1}^{N} \beta_k M_k \frac{I_j p_{jk}}{w_{hk}}. \tag{13.5}$$

The main challenge to implement one of those control protocols over a specific geospatial metapopulation network, as the one shown in Fig. 13.1, is the human mobility data between municipalities. However, some mathematical models as the gravity or radiation models [11] could be considered to assess the dwell-time parameters p_{ij}. The gravity model is based on the observation that the mobility between two patches i and j is proportional to the product of origin population N_{hi} and destination population N_{hj} and inversely proportional to a power law of the geographical distance r_{ij} between them. That is, in general [11]:

$$p_{ij} = A \frac{N_{hi} N_{hj}}{r_{ij}^{\delta}}, \tag{13.6}$$

where A is the normalization factor and δ are the parameters of the human mobility model to be calculated by multiple regression analysis. On the other hand, the radiation model is based on a diffusion model, where particles are emitted at a given location and, with a probability p, are absorbed by neighboring locations.

It is worth remarking that the SIR-SI model (13.3) assumes that the mosquito demography growth is constant. However, vector-borne disease, and particularly Dengue disease, is a seasonal disease outbreak mainly due to mosquitos that have favorable and unfavorable periods of time over a given year. Even more, some other factors as the climatological ones enhance the reproduction rate of mosquitoes when the temperature and the rain favor the habitat and the ecosystem of the mosquito population. The usual form of include such effect over a SIR-SI model, is by considering demographic parameters (recruitment rate Λ_k or mortality rate μ_{vk}) to be a time depend with a periodic function as the $\sin(\cdot)$ or $\cos(\cdot)$. For example, in [12] the authors consider that the mosquitos birth rate $\mu_{vk} N_{vk}$ is modeled as a sinusoidal function with a distinct amplitude, which results in the varying vector population size over time (high in the summer and low in the winter). Based on this idea, the effect of seasonality over the adult female mosquitoes population in the SIR-SI model (13.1)–(13.2) (or model (13.3)) is introduced as

$$\hat{\mu}_{vk}(t) = \mu_{vk}\left(1 - \epsilon sin\left(\frac{2\pi t}{365}\right)\right),$$

where $\epsilon > 0$ represents the amplitude of oscillations in vector birth rate.

Another alternative is to consider the mosquito recruitment rate Λ_k a type-dependent parameter given by Velázquez-Castro et al. [13]:

$$\Lambda_k(t) = \mu_{vk}\, N_{vk} \left(1 - \frac{N_{vk}}{C_k}\right), \tag{13.7}$$

that is, mosquito population exhibits logistic growth, with C_k the carrying capacity of a given patch k, which represent the environment conditions that modulated

the mosquito growth rate when resources increase or become limited. It is worth mentioning that the entire biological life cycle of the mosquito, from egg, larva, and pupa to adulthood, is compressed in the parameter $\Lambda_k(t)$. Even more, some dependencies of climatological variables as temperature or rainfall could be included in Λ_k through the parameter μ_{vk} and C_k.

An alternative to this seasonality model into a SIR-SI model (13.3) is by means of seasonal succession. The idea is to define two or more growth dynamics at different time intervals. Each time interval characterizes a particular season, for example winter or summer, where mosquito population has different demographic behavior. That is, a hybrid system for the SIR-SI model (13.3) is proposed with a time-dependent switching signal. For example, in the context of mathematical ecology, where P represents the size of a given species population, the authors in [14] analyze a model that switches between a logistic growth and an exponential negative growth given by

$$\begin{cases} \dot{P}(t) = -\lambda P(t)\,, & mw \le t \le mw + (1-\phi)w\,, \\ \dot{P}(t) = \alpha P(t)\Big(1 - \dfrac{P(t)}{C}\Big)\,, & mw + (1-\phi)w \le t \le (m+1)w\,; \end{cases} \tag{13.8}$$

where $\lambda > 0, m \in \mathbb{N}, \alpha$ is the intrinsic growth, C the environment carrying capacity, and $\phi \in (0, 1]$. It is worth noting that the system (13.8) alternate between season 1 during $[mw, mw + (1-\phi)w]$ and the season 2 during $[mw + (1-\phi)w, (m+1)w]$, and then, when the environment change, the population demographic dynamics turn back to season 1.

Observations from laboratory on adult *Aedes. aegypti*, have been assessing that below 14–15 °C, mosquitoes experience reduced mobility and that it cannot survive lower than 2–3 days without a blood meal [15]. Such factors increase the mosquitoes mortality rate; however, mosquitoes eggs are able to undergo diapause allowing the species to persist during cold winter temperature that are unfavorable to adult survival. In this context, the above change on the dynamical behavior of mosquitoes population growth could be included in the metapopulation network model (13.3) by introducing a piecewise constant signal $\sigma_i(t)$ taking value from an index set $\mathcal{Q} = \{1, 2\}$, and the individual modes set $\{\Lambda_{qi} : q \in \mathcal{Q}\}$, with two possible population dynamics $\Lambda_{1i} = -\lambda_i N_{vi}$ and $\Lambda_{2i} = \alpha_{vi} N_{vi}(1 - N_{vi}/C_i)$, with α_{vi} the mosquito intrinsic growth.

With these elements, the metapopulation network model for Dengue disease with seasonal succession is written as follows:

$$\text{Patch } k \begin{cases} \dot{S}_k &= \mu_{hk}(N_h - S_k) \quad - \sum_{j=1}^{N} \beta_j V_j \dfrac{S_k p_{kj}}{w_{hj}}\,, \\ \dot{I}_k &= -(\gamma_k + \mu_{hk}) I_k \quad + \sum_{j=1}^{N} \beta_j V_j \dfrac{S_k p_{kj}}{w_{hj}}\,, \\ \dot{R}_k &= \gamma_k I_k - \mu_{hk} R_k\,, \\ \dot{M}_k &= \Lambda_{\sigma(t)}^{(k)} - \mu_{vk} M_k \quad - \sum_{j=1}^{N} \beta_k M_k \dfrac{I_j p_{jk}}{w_{hk}}\,, \\ \dot{V}_k &= -\mu_{vk} V_k \qquad\quad + \sum_{j=1}^{N} \beta_k M_k \dfrac{I_j p_{jk}}{w_{hk}}\,; \end{cases} \tag{13.9}$$

with $k = 1, \ldots, N$ and

$$\sigma(t) = \begin{cases} 1\,, & if \quad mw \le t \le mw + (1-\phi)w\,, \\ 2\,, & if \quad mw + (1-\phi)w \le t \le (m+1)w\,; \end{cases} \tag{13.10}$$

In the following section a system of two connected patches is numerically analyzed as an example.

13.3 Numerical Example for Two Connected Patches

Consider a system of two connected patches as can be seen in Fig. 13.3. In specific, assume that the first patch corresponds to the municipality of Acapulco and that the second patch corresponds to the municipality of Coyuca de Benítez; both located at the state of Guerrero, in southwestern México. According to the 2020 Census of Population and Housing [16], Acapulco is inhabited by $N_{h1} = 779{,}566$ residents, and Coyuca de Benítez by $N_{h2} = 73{,}056$ residents. The geographical distance per road between these two cities is $r_{12} = 32.4$ km; using the gravity model (13.6), with $\delta = 2$ and $\mathrm{A} = 1 \times 10^{-14}$, the mobility between patch 1 (Acapulco) and patch 2 (Coyuca de Benítez) is $p_{12} = p_{21} = 0.598$ (it is worth remarking that for this

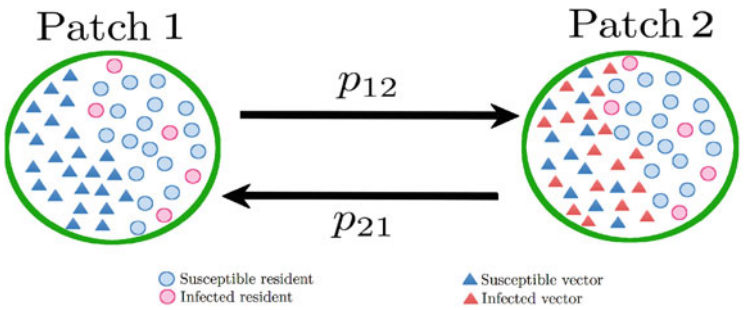


Fig. 13.3 A schematic representation of two patches connected by human mobility. As an example, it is assumed that the second patch is also inhabited by infected mosquitos at the beginning of the summer season. Here, the first patch represents the municipality of Acapulco, México, with $N_{h1} = 779{,}566$ residents, and the second patch the municipality of Coyuca de Benítez, México, with $N_{h2} = 73{,}056$ residents

example a symmetric mobility is considered.). Then, the fraction of the day that residents spend in its own patch is $p_{11} = 1 - p_{12}$ and $p_{22} = 1 - p_{21}$ for patch 1 and 2, respectively.

Then, the SIR-SI model with human mobility between two patches is expressed as

$$\text{Patch 1}\begin{cases} \dot{S}_1 &= \mu_{h1}(N_{h1} - S_1) - \beta_1 V 1 \dfrac{S_1 p_{11}}{w_{h1}} - \beta_2 V_2 \dfrac{S_1 p_{12}}{w_{h2}}\,, \\ \dot{I}_1 &= -(\gamma_1 + \mu_{h1}) I_1 + \beta_1 V 1 \dfrac{S_1 p_{11}}{w_{h1}} + \beta_2 V_2 \dfrac{S_1 p_{12}}{w_{h2}}\,, \\ \dot{R}_1 &= \gamma_1 I_1 - \mu_{h1} R_1\,, \\ \dot{M}_1 &= \Lambda^{(1)}_{\sigma(t)} - \mu_{v1} M_1 - \beta_1 M_1 \dfrac{I_1 p_{11}}{w_{h1}} - \beta_1 M_1 \dfrac{I_2 p_{21}}{w_{h1}}\,, \\ \dot{V}_1 &= -\mu_{v1} V_1 \qquad\quad + \beta_1 M_1 \dfrac{I_1 p_{11}}{w_{h1}} + \beta_1 M_1 \dfrac{I_2 p_{21}}{w_{h1}}\,; \end{cases} \tag{13.11}$$

and

$$\text{Patch 2}\begin{cases}\dot{S}_2 &= \mu_{h2}(N_{h2}-S_2)-\beta_1 V_1\dfrac{S_2 p_{21}}{w_{h1}}-\beta_2 V_2\dfrac{S_2 p_{22}}{w_{h2}}\,,\\ \dot{I}_2 &= -(\gamma_2+\mu_{h2})I_2+\beta_1 V_1\dfrac{S_2 p_{21}}{w_{h1}}+\beta_2 V_2\dfrac{S_2 p_{22}}{w_{h2}}\,,\\ \dot{R}_2 &= \gamma_2 I_2-\mu_{h2}R_2\,,\\ \dot{M}_2 &= \Lambda^{(2)}_{\sigma(t)}-\mu_{v2}M_2-\beta_2 M_2\dfrac{I_1 p_{12}}{w_{h2}}-\beta_2 M_2\dfrac{I_2 p_{22}}{w_{h2}}\,,\\ \dot{V}_2 &= -\mu_{v2}V_2+\beta_2 M_2\dfrac{I_1 p_{12}}{w_{h2}}+\beta_2 M_2\dfrac{I_2 p_{22}}{w_{h2}}\,;\end{cases} \tag{13.12}$$

with $w_{h1}=p_{11}N_{h1}+p_{21}N_{h2}$ and $w_{h2}=p_{22}N_{h2}+p_{12}N_{h1}$. In order to simplify the numerical example, assume that due to the geographic closeness between the two patches, both, the demographic of human population and mosquito entomological parameters, are equal, that is: $\mu_{h1}=\mu_{h2}=0.0000457$; $\gamma_1=\gamma_2=1/7$; $\beta_1=\beta_2=0.67$; and $\mu_{v1}=\mu_{v2}=1/8$. Furthermore, consider that the climate conditions are similar in both patches, such that summer and autumn seasons occur at the same period of time.

To define the switching times in (13.10), select, as an example, $m=0.01$, $w=99$, and $\phi=0.353$, such that the switching signal becomes

$$\sigma(t)=\begin{cases}1\,, & if \quad 1\le t\le 65\,, & \to summer,\\ 2\,, & if \quad 65\le t\le 100\,, & \to autumn.\end{cases} \tag{13.13}$$

In this context, the growth of susceptible mosquito in the patch $i\in\{1,2\}$ and for each season is given by

$$\Lambda^{(i)}_k=\begin{cases}\alpha_{vi}N_{vi}(t)\Big(1-\dfrac{N_{vi}(t)}{C_i}\Big)\,, & if \quad k=1\,, & \to summer,\\ -\lambda_i N_{vi}(t)\,, & if \quad k=2\,, & \to autumn;\end{cases} \tag{13.14}$$

where the per-capita mortality rate of adult female mosquitoes in autumn is given by $\lambda_1=\lambda_2=100$, and a number of eggs laid per day for every female mosquito in summer as $\alpha_{v1}=\alpha_{v2}=5$. It is worth remarking that, for mosquito-borne diseases, carrying capacity C_i is usually associated with the maximum number of adult mosquitoes that can be produced and sustained in the local ecosystem in a given season [17]. This parameter could be also correlated with the environmental pressures generated by human activities as the urbanization and agriculture, or by

the domestic water storage in large open drums or the accumulation of water-retaining waste products such as plastic containers and tin cans. In this context, two scenarios are considered in the numerical simulation: (a) both patches share the same environment conditions such that $C_1 = C_2 = 10{,}000$, or (b) the environment conditions in patch 2 favor mosquito productivity $C_2 = 200{,}000$ compared with patch 1 ($C_1 = 10{,}000$). In both scenarios, it is assumed that, at the beginning of the summer, patch 2 has 100 infected mosquitos (that could have come from diapaused eggs that hatched when they found the ideal environmental conditions). That is, the initial condition is ($S_1(0) = 779{,}566$; $I_1(0) = 0$; $R_1(0) = 0$; $M_1(0) = 9750$; $V_1(0) = 0$) for patch 1, and ($S_2(0) = 73{,}056$; $I_2(0) = 0$; $R_2(0) = 0$; $M_2(0) = 9650$; $V_2(0) = 100$) for patch 2.

In Fig. 13.4 the numerical solution of system (13.11)–(13.12) for the first scenario with $C_1 = C_2$ is shown. Here, a first period of 30 days without human mobility is considered at the beginning of the summer season, that is, $p_{12} = p_{21} = 0$ for $0 \leq t < 30$. As can be seen, the number of infected humans in patch 1 grows as soon the mobility between the two patches is considered; and as a consequence, the abundance of infected mosquitoes in patch 1 begins also to increase until the autumn season, where the curve starts to decrease.

On the other hand, in Fig. 13.4 the second scenario for the system (13.11)–(13.12) is illustrated, where the carrying capacity of the second patch is bigger than its neighboring patch. Before 30 days, where it has been assumed that mobility between patches is zero, the dengue outbreak in patch 2 occurs almost completely, presenting the typical curve of the infected residents. When the mobility starts, it is possible to observe a rapid growth of dengue disease in residents of patch 1, but with a small number of transmission to mosquitoes as can be seen in Fig. 13.5b.

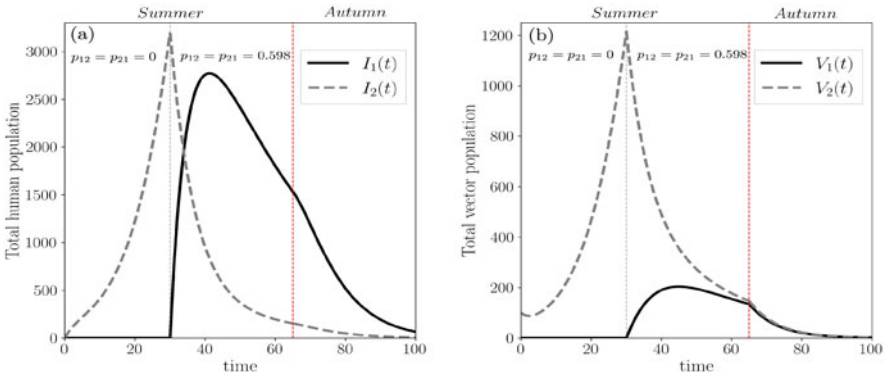


Fig. 13.4 Scenario 1, where $C_1 = C_2 = 10{,}000$. (**a**) The number of infected humans in both patches; and (**b**) the number of infected mosquitoes in both patches

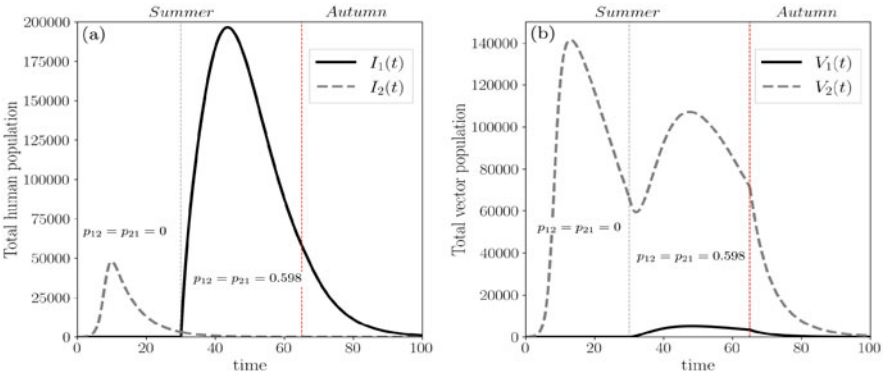


Fig. 13.5 Scenario 2, where $C_1 = 10{,}000$ and $C_2 = 200{,}000$. (**a**) The number of infected humans in both patches; and (**b**) the number of infected mosquitoes in both patches

13.4 The Role of Climate Variables on Mosquito Egg Diapause: Open Problems and Perspectives

Climate variables as temperature, rainfall, or humidity affect the mosquito life cycle and, consequently, generate the seasonality behavior in mosquito population dynamics [18, 19]. In particular, diapause phenomenon is a usually overlooked topic in the formulation of a mathematical model. It refers to an adaptive mechanism that inhibits the development of mosquito eggs in order to survive unfavorable weather conditions, making unable to hatch it. But, when temperature and lengths of photoperiod are favorable to the hatching of eggs, the mosquitoes population grows in such seasonal periods.

One possible methodology to include diapause phenomena in the geospatial metapopulation network model Eq. (13.3) is throughout the term Λ_{vk} in Eq. (13.14). Here diapause could be included by considering the parameter α_i (the number of mosquito eggs that hatch per day) as a piecewise function that separates, in a year, the favorable and unfavorable period. For example, in [20], by monitoring the presence and the relative abundance of the species, rate of positive ovitraps, and mean number of eggs per ovitrap, the authors have shown that in Rome, Italy, the first overwintering eggs in Rome hatched between the end of February and the beginning of March 2000, when—as the authors wrote—day length was 11–11.5 h of light and the mean temperature was 10–11 °C. In this sense we could say that for Rome, the favorable period for mosquitoes hatch is on the previously cited months.

To assess the distribution of *Aedes aegypti* in México, it is possible to get data from the network made up of 250,000 ovitraps distributed throughout the national territory and administrated by the SIMV (Sistema Integral de Monitoreo de Vectores) in [21]. On the other hand, in [22], the Organismos de Cuenca y Direcciones Locales de la CONAGUA provide a data set of 5500 weather stations

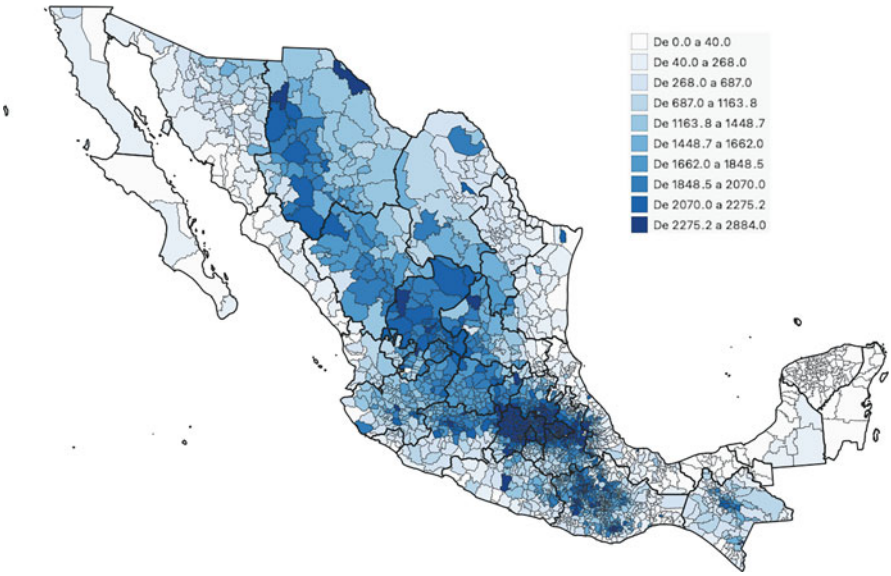


Fig. 13.6 Elevation of each México municipality, highlighting geographical regions with an elevation greater than 2000 m, where *Aedes aegypti* mosquito lives

(WS) distributed throughout the Mexican territory, with approximately 55 million daily records of rain in 24 hours and minimum and maximum temperature. In this direction, a *python* code to get and organize the WS data is implemented in order to know its geographical distribution and the period of years where there is a major abundance of WS with available data over all the municipalities of Mexico. From a first global statistical analysis, it is observed a downward trend in the number of stations with available data is clear, with 2019 being the lowest number of stations (with 354 stations), 2018 (with 1376 stations), and 2017 (with 2048 stations).

The geographical elevation of each México municipality is an important factor for Dengue analysis since it is well known that *Aedes aegypti* mosquito lives in regions with a maximum elevation of 2000 m [23]. Then, using INEGI National Geostatistical Framework [16], a first zoning risk region is shown in Fig. 13.6 where *Aedes aegypti* is predicted to occur within each elevation range per municipality. The aim, for future research works, is to include to this map the climate data from the WS, and other socio-economical variables, available in the INEGI population census 2020, to construct a risk zoning map for Dengue disease in México.

13.5 Conclusion

The methods of switched dynamical systems applied to the analysis and modeling of Dengue disease outbreaks in Mexico provide a valuable opportunity to address a relevant social problem. In this work a mathematical model based on a geospatial

metapopulation network and a compartmental SIR-SI epidemic model for vector-borne diseases with seasonal succession is proposed to analyze Dengue disease. Here, each node of the network, also called patch, represents a geographical territory as a city or an urban/rural locality, and the connection between patches represents the human mobility. As an example, the case of two patches is numerically analyzed, where, using the INEGI population census 2020 for the municipality of Acapulco and Coyuca de Benítez, in Guerrero, México, is studied. The gravity model to emulate the human mobility between such patches is also used to analyze the Dengue disease between those two patches. The model includes the succession of two seasons: a favorable period where the mosquito population grows according to a logistic model (where the carrying capacity parameter is considered to emulate the productivity of vector in a given ecosystem); and an unfavorable period in which the population behavior of mosquitos switches to an exponential decrease dynamics. Considering that, at the beginning of the summer the second patch is inhabited by infected mosquitos, it has been observed that the carrying capacity parameter affects the Dengue outbreak over the first patch, mainly when the carrying capacity of the second patch is bigger than its neighbor. Such scenario could model a situation where the economic or social factors in the second patch, as the accumulation of water-retaining waste, are worst in a given geographical region. Finally, a set of open problems that could improve the model are proposed, where diapause phenomena is also highlighted as a relevant phenomena that could be included in the model and completed with data from ovitraps; and the use of weather stations data to characterize favorable and unfavorable time periods for hatching of mosquito eggs.

Acknowledgments Authors acknowledgment CONACYT support through the program "Investigadoras e Investigadores por México (Cátedras CONACYT para Jóvenes Investigadores 2016") proyect 278. The author thankfully acknowledge the computer resources, technical expertise and support provided by the Laboratorio Nacional de Supercómputo del Sureste de México, CONACYT member of the network of national laboratories, project 202104091C.

References

1. J.P. Messina, O.J. Brady, N. Golding, M.U.G. Kraemer, G.R.W. Wint, S.E. Ray, D.M. Pigott, F.M. Shearer, K. Johnson, L. Earl, L.B. Marczak, S. Shirude, N. Davis Weaver, M. Gilbert, R. Velayudhan, P. Jones, T. Jaenisch, T.W. Scott, R.C. Reiner, S.I. Hay, The current and future global distribution and population at risk of dengue. Nat. Microbiol. **4**(9), 1508–1515 (2019)
2. J.P. Messina, O.J. Brady, D.M. Pigott, J.S. Brownstein, A.G. Hoen, S.I. Hay, A global compendium of human dengue virus occurrence. Sci. Data **1**(1), Article number: 14000 (2014)
3. Panorama Epidemiológico de Dengue 2019; Sistema Especial de Vigilancia Epidemiológica de Dengue de la Dirección de Vigilancia Epidemiológica de Enfermedades Transmisibles. https://www.gob.mx/salud/documentos/panorama-epidemiologico-de-dengue-2019
4. D. Bichara, C. Castillo-Chavez, Vector-borne diseases models with residence times–A Lagrangian perspective. Math. Biosci. **281**, 128–138 (2016)
5. C.A. Manore, K.S. Hickmann, J.M. Hyman, I.M. Foppa, J.K. Davis, D.M. Wesson, C.N. Mores, A network-patch methodology for adapting agent-based models for directly transmitted disease to mosquito-borne disease. J. Biol. Dyn. **9**(1), 52–72 (2015)

6. A. Anzo-Hernández, B. Bonilla-Capilla, J. Velázquez-Castro, M. Soto-Bajo, A. Fraguela-Collar, The risk matrix of vector-borne diseases in metapopulation networks and its relation with local and global R0. Commun. Nonlinear Sci. Num. Simul. **68**, 1–14 (2019)
7. S.T. Stoddard, A.C. Morrison, G.M. Vazquez-Prokopec, V. Paz Soldan, T.J. Kochel, U. Kitron, J.P. Elder, T.W. Scott, The role of human movement in the transmission of vector-borne pathogens. PLoS Negl. Trop. Dis. **3**(7), e481 (2009)
8. J. Reinhold, C. Lazzari, C. Lahondère, Effects of the environmental temperature on aedes aegypti and aedes albopictus mosquitoes: a review. Insects **9**(4), 158 (2018)
9. Y. Lou, K. Liu, D. He, D. Gao, S. Ruan, Modelling diapause in mosquito population growth. J. Math. Biol. **78**(7), 2259–2288 (2019)
10. R.A. Erickson, S.M. Presley, L.J.S. Allen, K.R. Long, S.B. Cox, A dengue model with a dynamic Aedes albopictus vector population. Ecol. Modell. **221**(24), 2899–2908 (2010)
11. A.P. Masucci, J. Serras, A. Johansson, M. Batty, Gravity versus radiation models: on the importance of scale and heterogeneity in commuting flows. Phys. Rev. E **88**(2), 022812 (2013)
12. J.E. Kim, H. Lee, C.H. Lee, S. Lee, Assessment of optimal strategies in a two−patch dengue transmission model with seasonality. PLOS ONE **12**(3), e0173673 (2017)
13. J. Velázquez-Castro, A. Anzo-Hernández, B. Bonilla-Capilla, M. Soto-Bajo, A. Fraguela-Collar, Vector-borne disease risk indexes in spatially structured populations. PLOS Negl. Trop. Dis. **12**(2), e0006234 (2018)
14. S.-B. Hsu, X.-Q. Zhao, A Lotka–Volterra competition model with seasonal succession. J. Math. Biol. **64**(1–2), 109–130 (2011)
15. O.J. Brady, M.A. Johansson, C.A. Guerra, S. Bhatt, N. Golding, D.M. Pigott, H. Delatte, M.G. Grech, P.T. Leisnham, R. Maciel-de-Freitas, L.M. Styer, D.L. Smith, T.W. Scott, P.W. Gething, S.I. Hay, Modelling adult Aedes aegypti and Aedes albopictus survival at different temperatures in laboratory and field settings. Parasites Vectors **6**(1), 1–12 (2013)
16. Instituto Nacional de Estadística y Geografía (INEGI). www.inegi.org.mx
17. D.L. Smith, T.A. Perkins, L.S. Tusting, T.W. Scott, S.W. Lindsay, Mosquito population regulation and larval source management in heterogeneous environments. PLoS ONE **8**(8), e71247 (2013)
18. A. Abdelrazec, A.B. Gumel, Mathematical assessment of the role of temperature and rainfall on mosquito population dynamics. J. Math. Biol. **74**(6), 1351–1395 (2016)
19. P. Jia, L. Lu, X. Chen, J. Chen, L. Guo, X. Yu, Q. Liu, A climate–driven mechanistic population model of *Aedes albopictus* with diapause. Parasites Vectors **9**(1), 1–15 (2016)
20. L. Toma, F. Severini, M.D. Luca, A. Bella, R. Romi, Seasonal patterns of oviposition and egg hatching rate of Aedes albopictus in Rome. J. Am. Mosq. Control Assoc. **19**(1), 19–22 (2003)
21. Sistema Integral de Monitoreo de Vectores, http://kin.insp.mx/aplicaciones/SisMV/Home/SisMV
22. Organismos de Cuenca y Direcciones Locales de la CONAGUA. https://smn.conagua.gob.mx/es/climatologia/informacion-climatologica/informacion-estadistica-climatologica
23. A.G. Watts, J. Miniota, H.A. Joseph, O.J. Brady, M.U.G. Kraemer, A.W. Grills, S. Morrison, D.H. Esposito, A. Nicolucci, M. German, M.I. Creatore, B. Nelson, M.A. Johansson, G. Brunette, S.I. Hay, K. Khan, M. Cetron, Elevation as a proxy for mosquito-borne Zika virus transmission in the Americas. PLOS ONE **12**(5), e0178211 (2017)

Index

G. Huerta Cuellar et al. (eds.), *Complex Systems and Their Applications*,
https://doi.org/10.1007/978-3-031-02472-6