REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

ECOLE NATIONALE POLYTECHNIQUE

Department of Control Engineering

Process Control Laboratory

A Thesis Submitted In Fulfillment of The Requirements
For The Degree of State Engineer

Chaos synchronization using non-linear observers with applications to cryptography

Author:

Mohamed Camil
BELHADJOUDJA

Supervisors:

Prof. Mohamed TADJINE
Dr. Messaoud CHAKIR

Jury

President	Mr. M.S. BOUCHERIT	Professor ENP
Examiner	Mr. R. ILLOUL	MCA ENP
Supervisor	Mr. M. TADJINE	Professor ENP
Supervisor	Mr. M. CHAKIR	Doctor ENP

ENP 2021

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

ECOLE NATIONALE POLYTECHNIQUE

Department of Control Engineering

Process Control Laboratory

A Thesis Submitted In Fulfillment of The Requirements
For The Degree of State Engineer

Chaos synchronization using non-linear observers with applications to cryptography

Author:

Mohamed Camil
BELHADJOUDJA

Supervisors:

Prof. Mohamed TADJINE
Dr. Messaoud CHAKIR

Jury

President	Mr. M.S. BOUCHERIT	Professor ENP
Examiner	Mr. R. ILLOUL	MCA ENP
Supervisor	Mr. M. TADJINE	Professor ENP
Supervisor	Mr. M. CHAKIR	Doctor ENP

ENP 2021

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

ECOLE NATIONALE POLYTECHNIQUE

Département d'Automatique

Laboratoire de Contrôle des Processus

Mémoire de projet de fin d'études
pour l'obtention du diplôme d'ingénieur d'état en Automatique

Synchronisation du chaos à base d'observateurs non-linéaires avec des applications en cryptographie

Auteur:
Mohamed Camil
Belhadjoudja

Encadrants:
Prof. Mohamed Tadjine
Dr. Messaoud Chakir

Jury

Président	Mr. M.S. BOUCHERIT	Professeur ENP
Examinateur	Mr. R. ILLOUL	MCA ENP
Encadrant	Mr. M. TADJINE	Professeur ENP
Encadrant	Mr. M. CHAKIR	Docteur ENP

ENP 2021

ملخص

موضوع هذه الأطروحة هو تزامن الأنظمة الفوضوية باستخدام مراقبين غير خطيين ، مع تطبيقات للتشفير. أقدم في رسالتي طريقة عامة لتحسين أداء بعض المراقبين ذوي المدخلات غير المعروفة. كما أنني أطور طريقة لتحسين خصائص التردد للأنظمة الفوضوية مثل نظام روسليرس ونظام لورنز. أقوم أيضًا ببناء تعديلات على مراقب التواء للأنظمة مع تأخير الوقت. من بين هذه التعديلات ، هناك طريقة تتيح لنا اعتبار أي مدخلات غير معروفة لنظام ما في شكل مثلث كحالة وهمية ، مما يسمح لنا باستخدام مراقبي الحالة لتقدير مدخلات غير معروفة. أقوم أيضًا ببناء فكرة مراقبي الوضع الانزلاقي ذو الترتيب التكيفي العالي. أقدم أيضًا إجراءً مضادًا باستخدام مراقبي ANFIS ونظام لورنز المحسن الذي يواجه هجوم تحليل الشفرات الصوتي ضد RSA ، ثم قمت بدمج نظرية الفوضى مع نظرية الشبكة لتوليد مشاكل جديدة في تشفير ما بعد الكم.

كلمات مفتاحية :

نظرية الفوضى ، التشفير ، المراقبون غير الخطيين ، تزامن الفوضى ، المشابك.

RESUME

Le sujet de ce mémoire est la synchronisation des systèmes chaotiques à l'aide d'observateurs non linéaires, avec des applications en cryptographie. Je présente dans mon mémoire une méthode générale pour améliorer les performances de certains observateurs à entrées inconnues. Je développe également une méthode pour améliorer les caractéristiques fréquentielles des systèmes chaotiques tels que le système de Rossler et le système de Lorenz. Je construis des modifications d'un observateur "Super Twisting" pour les systèmes avec retard dans la sortie. Je conçois une méthode qui permet de considérer toute entrée inconnue d'un système sous forme triangulaire comme état fictif, ce qui permet d'utiliser des observateurs d'état pour l'estimation d'une entrée inconnue. Je construis égallement la notion d'observateurs à mode glissant d'ordre supérieur adaptatif. Je présente une contre-mesure utilisant des observateurs ANFIS et un système de Lorenz amélioré face à une technique de cryptanalyse acoustique contre RSA. Pour finir, je combine la théorie du chaos avec la théorie des réseaux Euclidiens pour générer de nouveaux problèmes en cryptographie post-quantique.

Mots clés : Théorie du Chaos, Cryptographie, Observateurs non linéaires, Synchronization du chaos, Réseaux Euclidiens.

ABSTRACT

The subject of this thesis is the synchronization of chaotic systems using non-linear observers, with applications to cryptography. I present in my thesis a general method to improve the performance of some observers with unknown inputs. I also develop a method to improve the frequency characteristics of chaotic systems such as the Rossler system and the Lorenz system. I also build modifications of a twisting observer for systems with time delay. Among these modifications, there is a method that allows us to consider any unknown input of a system in triangular form as being a fictitious state, which allows us to use state observers for the estimation of an unknown input. I construct also the notion of higher-adaptive-order sliding mode observers. I also present a countermeasure using ANFIS observers and an improved Lorenz system facing an acoustic cryptanalysis attack against RSA, then I combine the chaos theory with the lattice theory to generate new problems in post-quantum cryptography.

Key words: Chaos theory, Cryptography, Nonlinear observers, Chaos synchronization, Lattices.

“

Je dédie ce travail à Mamy Sacia, Papy M'hamed, Mamy Khadra, Mamy Frida et souhaite longue vie et santé à Papy Mohand Said

”

- Camil

Acknowledgments

Acknowledgments are probably the hardest part of writing a thesis. Because we must achieve the impossible: Express in few words our infinite gratitude to people who have made us who we are today.

My first thanks go to those dearest to me: my family. My mom, dad, sister, grandparents, uncles, and aunts. They are my strength, my courage, and my ambition. Without them, I would be nothing. They are my inexhaustible source of happiness and the only thing that really makes sense.

I would also like to thank my supervisors, Mr. Tadjine and Mr. Chakir for their invaluable advice and the intellectual and moral support they gave me during the writing of my thesis.

I also want to thank Feynman, Galois, Bourbaki, Alain Connes, Cedric Villani, and so many others for teaching me what it really means to be a scientist.

I would like to thank my teacher of Algebra Mr. Ouadjaout, for being the first to put me on the path of pure mathematics. And my Analysis teachers Mr. Mahdi and Mr. Kebli, for showing me the beauty of applied mathematics.

Finally, I want to thank my friends for their encouragement and the unforgettable moments I shared with them.

Table of Contents

List of Figures

NOMENCLATURE

$\dot{z}$	First time derivative
$\exists$	Exists
$\forall$	For all
$\frac{\partial}{\partial x}$	Partial derivative with respect to x
$\int_a^b$	Definite integral
λ_{max}	Largest singular value
λ_{min}	Smallest singular value
$\mathbf{Z}/m\mathbf{Z}$	The ring of integers modulo m
A^+	Generalized inverse of A
A^T	Transpose of A
$L_f h$	Lie derivative of h along f

CHAPTER 1

INTRODUCTION

1.1 Context and motivations

The subject of this thesis is the synchronization of chaotic systems based on non-linear observers for applications in cryptography.

Cryptography is the science of secure data transmissions. It is part, along with cryptanalysis -which is the science that aims to test the security of cryptographic systems (cryptosystems)- of the broad field of cryptology.

We essentially find two categories of techniques for the design of cryptosystems: mathematical techniques and physical techniques. The vast majority of mathematical techniques are related to number theory, elliptical curves, abstract algebra, and more recently the theory of lattices. The physical techniques are focused on the use of notions of quantum mechanics such as the Heisenberg principle and the polarization of photons.

Over the years, we have seen the emergence of a new mathematical technique for the design of cryptosystems, namely chaos theory. This emergence gave birth to what is today called chaotic cryptography, which is still at the experimental stage and which could in the coming years be a significant support to already existing encryption systems.

A chaotic system can be implemented in two different ways. In the form of a computer

program (C code for example) or in physical form (an electrical circuit). The problem with the computer implementation is that there is what is called the phenomenon of dynamic degradation. Because finite precision is used in a computer implementation, the program may behave differently from the actual chaotic system. For this reason, it is preferable to use a physical realization of the chaotic system based on electronic components.

In cryptography, there is an emitter that transmits the encrypted message and a receiver that decrypts it. In chaotic cryptography, to perform these encryption and decryption operations, we need to reproduce the same chaotic signal at the emitter and the receiver. A first idea would be to design the same electrical circuit with the same parameters. The problem with this idea is that it is impossible to reproduce with infinite precision the same circuits, there will always be uncertainty and noise which will cause differences between the parameters of the two chaotic circuits. And because of the strong dependence on the initial conditions of chaotic systems, the slightest difference, however small it may be, between the parameters of the two chaotic circuits will cause a large difference between the chaotic signals created at the level of the emitter and the level of the receiver. To overcome this, we use chaos synchronization.

Chaos synchronization consists of ensuring that a dynamical system called the slave system reproduces with a certain precision the signals emitted by a chaotic circuit called the master system. There are several chaos synchronization techniques. In my thesis, I work on techniques based on the theory of non-linear observers.

An observer is a dynamical system whose role is to reproduce certain signals based on partial information on the system which initially emitted these signals. The idea of synchronization based on non-linear observers is to take as the slave system an observer whose objective is to reproduce the signals emitted by a chaotic circuit.

1.2 Objectives of the thesis

The objectives of this thesis are as follows:

a. Provide techniques for improving chaotic synchronization methods based on the theory of non-linear observers.

b. Design a method of improving chaotic systems to obtain chaotic signals more suited to cryptography.

c. Provide a countermeasure based on chaos and non-linear observers against an attack called acoustic cryptanalysis.

d. Present ideas of chaotic cryptographic systems.

e. Design modifications, based on chaos theory, of optimization problems on lattices for uses in post-quantum cryptography.

1.3 Organisation of the thesis and contributions

Chapter 2 of this thesis is a state of the art. I start by presenting cryptography in general, then chaos theory, its application in cryptography and the theory of non-linear observers while highlighting the problem of chaos synchronization based on non-linear observers.

In chapter 3, I present two robust observers constructed in [12]. The first is an adaptive unknown inputs observer and the second is an adaptive sliding mode unknown inputs observer. I improve these observers by introducing techniques based on calculating the integral of the estimates of the unknown inputs. These techniques make it possible to reduce the estimation error of the unknown inputs. The problem is that they cause a delay. This delay is imposed by the mathematical structure of the improvement presented. These improvements can be generalized to all observers with unknown inputs as long as the inputs are binary or piecewise continuous. I also build two encryption systems using chaotic synchronized systems based on the improved versions of the observers, and I show through these examples that the delay is not necessarily a problem. I also build in this chapter a method allowing to improve the frequency characteristics of chaotic systems. Indeed, classical chaotic systems such as the Lorenz system or the Rossler system exhibit low-frequency characteristics, which generally makes it possible to break encryption systems based on these chaotic systems using filters. I show in this chapter that it is enough to noise in a controlled way the derivative of the state vector of a chaotic system to improve its frequency characteristics. I use this technique to build a range of modified Lorenz attractors whose frequency characteristics are better than those of the original Lorenz system.

In chapter 4, I present a predictor based super twisting second-order sliding mode observer built-in [18] for systems with a fixed and known delay in the output. I replace the sign functions of this observer with fuzzy inference systems to eliminate the chattering effect caused by the discontinuities of the sign function. This technique has already been used in articles such that [27, 28]. Then, I modify the problem by considering systems with unknown input. I show that for a system in triangular form, the addition of an unknown input can be considered as the addition of a fictitious state. It means that the state estimate allows, among other things, an unknown input estimation. I end this chapter by introducing the notion of higher-adaptive-order sliding mode observers by considering the case of the super twisting observer where the power is no longer 1/2 but a function that varies according to an optimization criterion.

In chapter 5, I construct a countermeasure against a cryptanalysis technique called acoustic cryptanalysis [16]. This cryptanalysis technique exploits the sound emitted by RSA decryption operation to reconstitute the secret key. My countermeasure consists of masking the sound emitted by RSA decryption by using a chaotic signal. The chaotic system that I use is one of the Lorenz attractors that I obtained with the method presented in chapter 3. To prevent a spy from deleting the chaotic signal used for masking using chaos synchronization techniques, I cascade the chaotic system with an ANFIS (Adaptive Neuro-Fuzzy Inference System) which is trained to reproduce white noise given a chaotic input.

In chapter 6, I propose two modifications of the SVP (Shortest vector problem) on lattices. SVP is a problem on lattices deemed difficult and resistant to quantum technology. My objective is to propose two new versions of the SVP whose security is unconditional, in the sense that it does not depend on the mathematical difficulty of solving this problem as is the case of the classical SVP. The first modification is to add a constraint that depends on the trajectories of a given chaotic system. The second modification consists of reducing the space of solutions of the SVP by making an intersection with the lattice and a given chaotic attractor. In either case, the initial parameters and conditions of the chaotic system used are kept secret, making the resolution of these problems by an attacker impossible as long as the initial parameters and conditions of the chaotic system are kept secret.

Part I

CHAPTER 2

STATE OF THE ART

2.1 Introduction

The goal of this chapter is to present an overview of research in chaotic cryptography. I start with an overview of cryptography. Then, I recall the basics of chaos theory and how to use chaotic systems for cryptography, with an introduction to the problem of chaos synchronization. Then, I present the theory of non-linear observers, which is used in this thesis for the synchronization of chaotic systems. I start with an explanation of the observability problem. Then, I introduce some of the classical observers: Kalman filter, Luenberger observer, Extended Kalman filter, Thau's observer, and High gain observer. I finish by introducing the more advanced observers: Adaptive observers, Unknown inputs observers, Sliding mode observers, Super twisting observers, and ANFIS (Adaptive Neuro-Fuzzy Inference Systems) observers.

2.2 Cryptography.

Cryptography is the science behind the design of secure communication systems. We find it for example in the military, political, industrial, and medical fields where there is a transmission of private data. The main goal of cryptography is to transform the data into a code that is not understandable by an eventual attacker. For example, in World War 2, the Nazis used a machine called ENIGMA for secure military communications. The problem with ENIGMA is that it had some weaknesses (for example, a letter

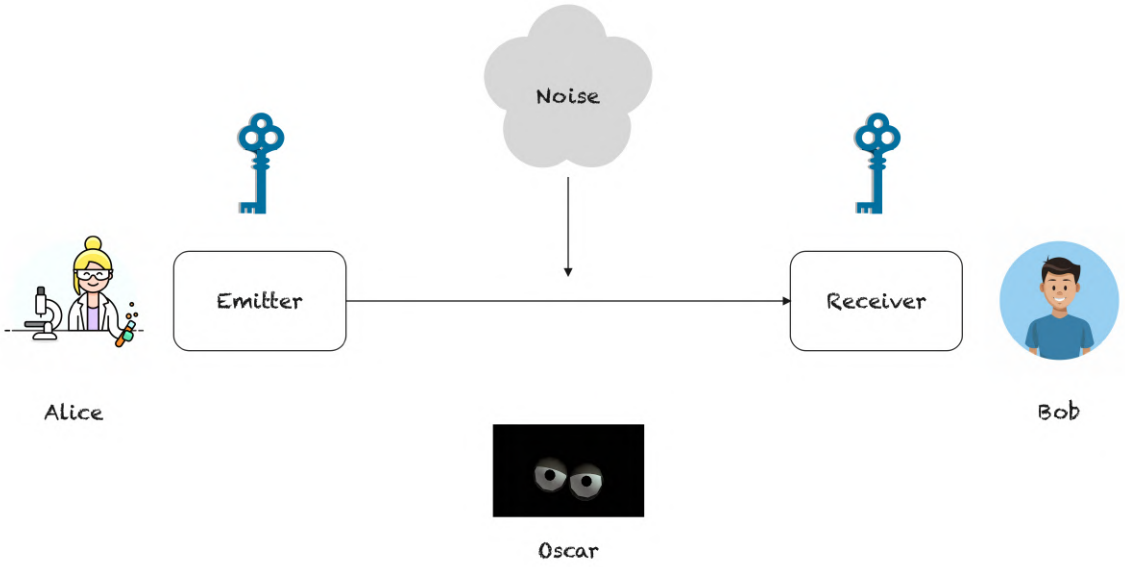


Figure 2.1: The structure of a cryptosystem

can be anything in the encrypted version except itself), which allowed the allies to break it. According to specialists, breaking ENIGMA shortened the war by at least 2 years. Cryptography is, therefore, an area of capital importance that it is essential to continuously develop.

2.2.1 Cryptosystems

The fundamental element in cryptography is the cryptosystem whose structure is shown in figure 2.1. Alice wants to send a message (the plaintext) to Bob but does not want this message to be read by a spy who will be called Oscar. To send her message, Alice uses an emitter which transforms the plaintext into a code (a ciphertext) and Bob uses a receiver to decode the ciphertext and read the message. To operate the emitter and receiver, Alice and Bob must use a key. The receiver's key is kept secret, in the sense that only Bob knows it. The emitter's key can be secret, and in this case, it is generally the same as Bob's and we talk about symmetric cryptography, or public, and in this case, we talk about asymmetric cryptography or public-key cryptography.

I will describe now a historical cryptosystem: The Shift Cipher, whose principle is shown in figure 2.2. The Shift Cipher is the cryptosystem that was used in ancient Rome by Caesar when he wanted to communicate with his generals. First, each letter is

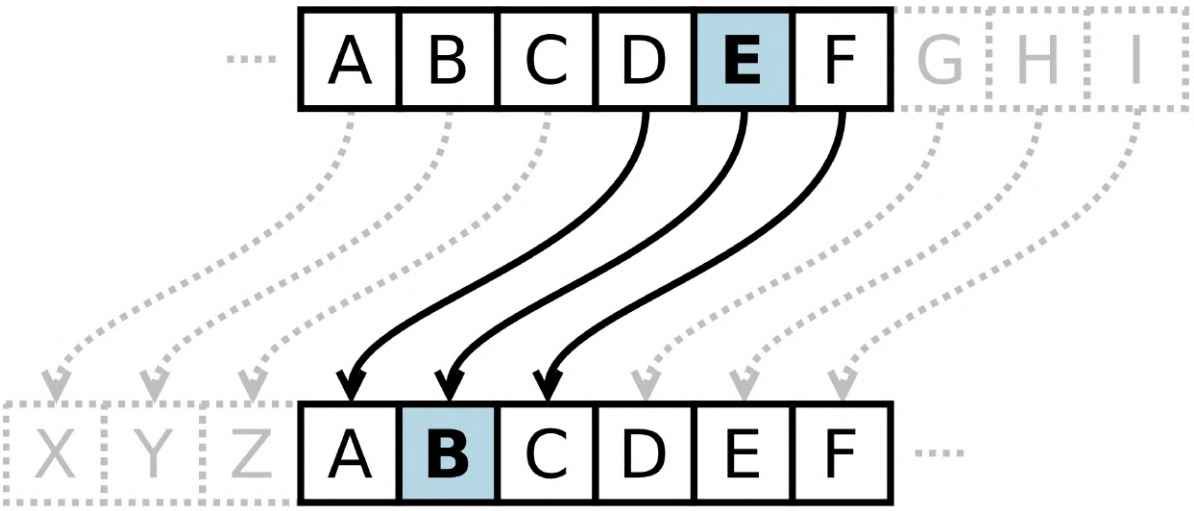


Figure 2.2: The Shift Cipher

associated with a number, and the plaintext is thus translated into a series of numbers. Then, we choose a key between 0 and 25, which must only be known by the two sides of the communication system. After that, each number of the plaintext is added modulo 26 to the key. The sequence obtained is translated into letters, which give us the ciphertext that can be sent. To decrypt the code, we transform the ciphertext into a series of numbers, and we subtract modulo 26 from each number the key, then we translate the result into a series of letters. The result is the plaintext. In the Shift Cipher, the emitter and receiver have the same key and it must be kept secret otherwise anyone will be able to easily understand the ciphertext.

I will introduce now cryptanalysis, the science that studies the security of cryptosystems. The fundamental idea when we design a cryptosystem is The Kerckhoffs principle. According to this principle, to have a secure communication system, it is necessary when we design it to assume that the attacker will know the encryption and decryption algorithm, except the secret key, and that he will have access to the emitter and the receiver. In other words, we should minimize the number of secret components in a cryptosystem, and we must not suppose that the architecture of the cryptosystem is secret[1].

To talk about the security of a cryptosystem, we must study the key space, plaintext space and ciphertext space. These terms are explained in the following definition:

[1] In general it is easy to find it using techniques like reverse engineering.

Definition 1 [25]: *Cryptosystem*
A cryptosystem is a five-tuple of sets $(\mathscr{P},\mathscr{C},\mathscr{K},\mathscr{E},\mathscr{D})$ *such that :*
1- $\mathscr{P}$ *is a finite set of possible* ***plaintexts****.*
2- $\mathscr{C}$ *is a finite set of possible* ***ciphertexts****.*
3- $\mathscr{K}$*, the* ***keyspace****, is a finite set of possible* ***keys****.*
4- For each $K \in \mathscr{K}$*, there is an* ***encryption rule*** $e_K \in \mathscr{E}$ *and a corresponding* ***decryption rule*** $d_K \in \mathscr{D}$*. Each* $e_k : \mathscr{P} \to \mathscr{C}$ *and* $d_K : \mathscr{C} \to \mathscr{P}$ *are functions such that* $d_K(e_K(x)) = x$ *for every plaintext element* $x \in \mathscr{P}$*.*

Let us now analyze the security of the shift cipher. The first method to break the shift cipher is the Brute force attack. We test several keys until we find the right one. In practice, in secure cryptosystems the keyspace is so large that it is almost impossible to find the right key in this way. However, in the case of the shift cipher, there are only 26 keys to test, which makes the brute force attack relatively effective.
The second method of cryptanalysis is frequency analysis. A frequency diagram of the appearance of letters in English is shown in figure 2.3. It can be used, with a diagram of the appearance of terms in English, to compare the frequency of appearance of letters and terms in the ciphertext with the frequency of appearance of letters and terms in English (or in general in the language used for communication). For example, the letter most used in English is E, we can then assume that the letter that is most used in the ciphertext corresponds to the encrypted version of E (as long as the ciphertext is sufficiently large).
Another useful method for basic cryptosystems such as the shift cipher is to choose a plaintext, get the associated ciphertext, change a single letter of the plaintext, and see how the ciphertext varies. For the shift cipher, by changing a single letter of the plaintext it will change a single letter of the ciphertext, we can therefore deduce that this letter is the encrypted version of the letter that we changed in the plaintext. This method, just as the frequency analysis attack, uses the weaknesses of the internal structure of the algorithm. It is classified as a mathematical analysis attack. There are many other attacks for different cryptosystems. We can use for example implementation attacks, which are techniques using the weaknesses of the hardware and software that are used. Another category of attacks is social engineering where we use the vulnerabilities of the human mind.

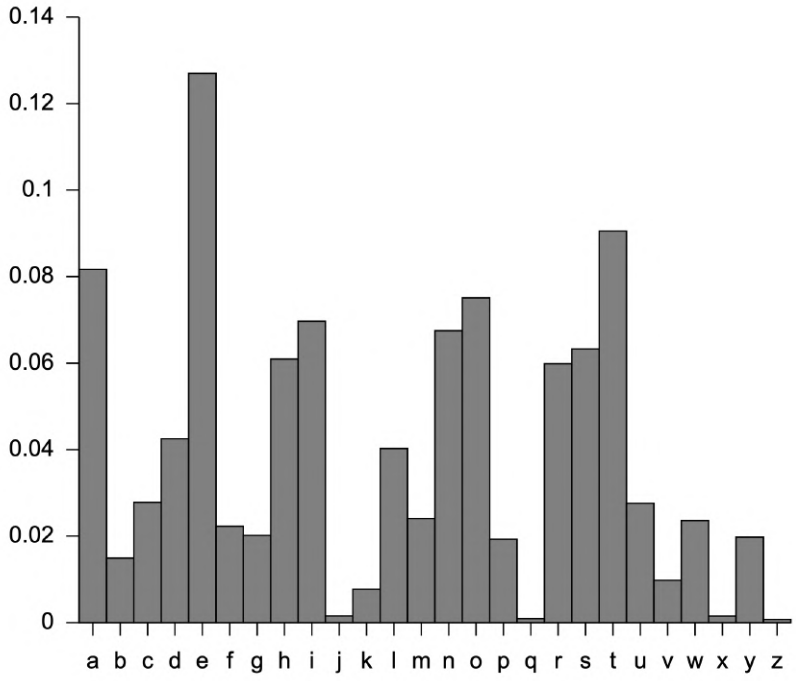


Figure 2.3: The frequency of appearance of letters in English

2.2.2 Symmetric cryptography

There are two types of symmetric cryptosystems. First, we have the Stream ciphers. The structure of a stream cipher is shown in figure 2.4. Stream ciphers are cryptosystems where we encrypt one bit each time and send it to the receiver for decryption. This operation is realized using XOR gates and bitstream generators. We XOR each bit of the plaintext with the bit generated by the bitstream generator, and then we do the same thing at the receiver side to recover the plaintext bit. The secret key is the input of the bitstream generator, and because we must have the same bitstream added at the receiver and the emitter side, we need the same key on the two sides, it is the reason why this cryptosystem is symmetric.

The bitstream generator needs to be random (TRNG: True Random Number Generator), or at least pseudo-random (PRNG: Pseudo-Random Number Generator). It also must have a fundamental property that ensures the security of this algorithm: unpredictability, which means that even if we know of any number of bits generated by the bitstream generator, it is computationally infeasible to compute the next (or preceding) bits from this information. These properties are here to make it impossible to guess the bits and decrypt the message without the key. An unpredictable PRNG is also called a CSPRNG

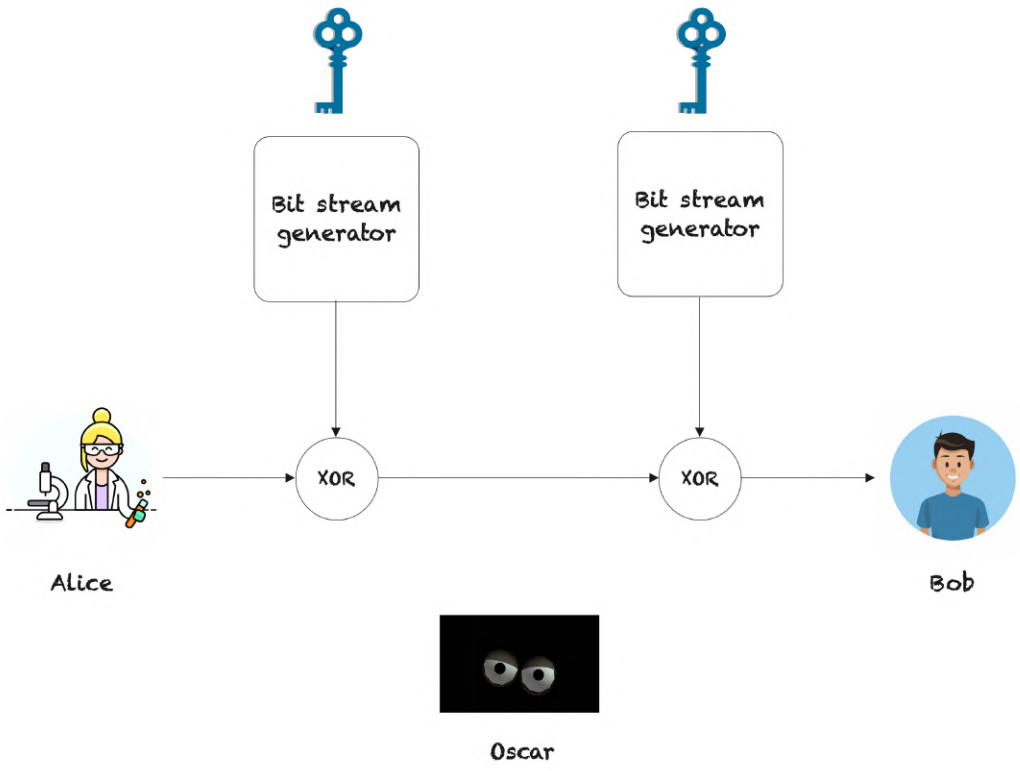


Figure 2.4: The structure of a stream cipher

(Cryptographically Secure Pseudo-Random Number Generator) [25]. The security of a stream cipher will depend mostly on the statistical properties of the CSPRNG used. We should ensure that the bitstream generator has good randomness properties and that it is unpredictable. There are mathematical tests of randomness like the Diehard tests and the chi-square test. There are also software like TestU01 that are dedicated to randomness tests. For unpredictability, we must ensure that the relationship between the bits generated by the bitstream generator is sufficiently complex.

The PRNG can be implemented into a CPU or it can be realized uniquely with hardware components. An example of a PRNG that is implemented in hardware is the Linear Feedback Shift Register (LFSR), whose structure is shown in figure 2.5.

In a LFSR, there are flip-flops, XOR gates, and multiplication symbols. The multiplication symbols are here to act as switches. If $p_i = 0$ then the switch is open, and if $p_i = 1$ the switch is closed. The bitstream is the sequence of outputs of the last flip-flop

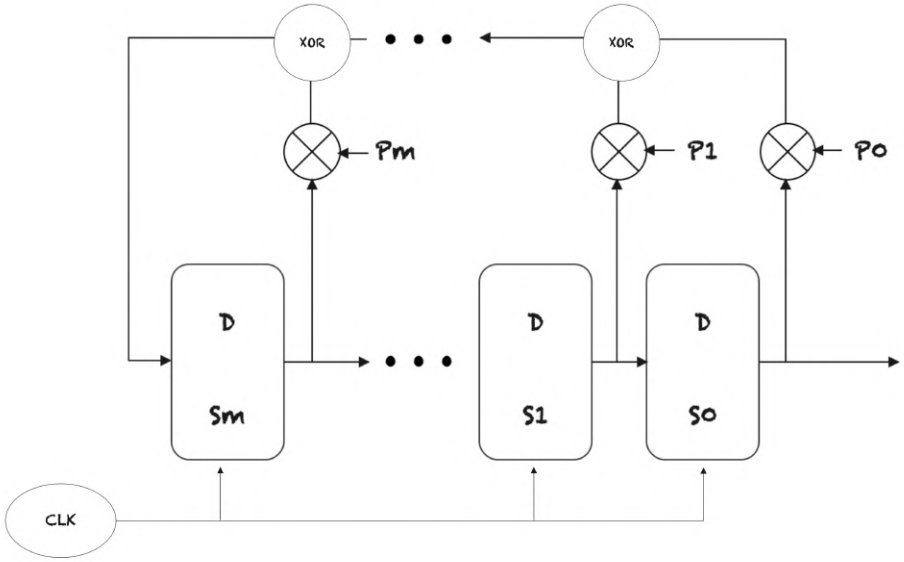


Figure 2.5: The structure of a LFSR

(s_0). The mathematical description of the input of the leftmost flip-flop is given by 2.1.

$$s_{m+1} \equiv s_m p_m + ... + s_1 p_1 + s_0 p_0 \ mod\ 2 \tag{2.1}$$

The maximum sequence length generated by this type of PRNG is $2^{m+1} - 1$, where $m+1$ is the number of flip-flops. The maximum length is however generated only by some combinations of p_is. For example, for $m+1 = 4$, $p_3 = p_2 = 0$ and $p_1 = p_0 = 1$, the PRNG has a sequence of maximum length, namely $2^4 - 1 = 15$.
The vector $(p_0, p_1, ..., p_{m-1}, p_m)$ is the secret key. It is common to represent it as a polynomial with coefficients in $\{0, 1\}$ as in 2.2.

$$P(x) = x^{m+1} + p_m x^m + p_{m-1} x^{m-1} + ... + p_1 x + p_0 \tag{2.2}$$

We can show that maximum-length PRNGs have primitive polynomials which can easily be computed. Thus, it is easy to find the maximum-length PRNGs.

It is clear that if we have $m+1$ different equations of the form 2.1, it will be easy to compute the secret key (because it will become a system of $m+1$ linear equations with $m+1$ unknowns). It is one of the weaknesses of LFSR-based stream ciphers. The bitstream generator is not unpredictable. One way of overcoming this is to make the p_is random (generating continuously random keys).

The other symmetric cryptosystems used in practice are Block ciphers. Block ciphers are the most secure and popular symmetric ciphers in cryptography. Unlike a stream cipher where we proceed one bit at a time, with block ciphers we encrypt a group of bits at once. The most popular block ciphers are AES (Advanced Encryption Standard), DES (Data Encryption Standard) and their modifications (3DES, ...). For more details about block ciphers, you can refer to [25]. I will not use block ciphers in this thesis. I have mentioned them in this chapter only to introduce the concepts of confusion and diffusion which are fundamental in cryptography [30]:

- **Confusion** : It is an encryption operation where the relation between the key and the ciphertext is obscured.
- **Diffusion** : It is also known as the avalanche effect. If we change one bit of plaintext, it must affect many ciphertext bits. It is done to mask the statistical properties of the plaintext.

2.2.3 Public key cryptography

Public key cryptography is a branch of cryptography where the emitter's key is public and the receiver's key is private. We call such cryptosystems asymmetric cryptosystems or public-key cryptosystems. Their main interest is that anyone can send a message, but only one person can decrypt it. It is for example used during banking transactions where anyone can send information to a bank but only the bank is able to read this information. I will not detail how public-key encryption algorithms work because they are not the ones that will be useful to us in this thesis. For details about the public key cryptosystems, you can refer to [25]. However, I will explain what the security of the most popular public-key algorithms is based on.

The most popular and possibly the most secure public-key cryptosystem is RSA (Rivest-Shamir-Adleman). The security of RSA is essentially based on the problem of factorization of integers which is stated as follows: Given an integer n, find the two primes p and q such that $n = pq$. It is considered to be a difficult problem, in the sense that no current computer could solve this problem in a suitable time. For current computers, the best published algorithm for integer factorization is **GNFS** (general number field sieve) that runs on a b-bit number n in time: $exp(((64/9)^{1/3} + o(1))(ln\ n)^{1/3}(ln\ ln\ n)^{2/3})$.

Another public-key algorithm is the Diffie-Hellman key exchange. It is a public key

algorithm that allows Alice and Bob to create the same secret key that they can use in a symmetric encryption algorithm. The security of the Diffie-Hellman key exchange is based on the discrete logarithm problem which is stated as follows: Given an integer t and a generator g of $\mathbf{Z}/m\mathbf{Z}$, compute $l = log_g t$. The discrete logarithm problem is considered, like the factorization of integers, to be a difficult problem.

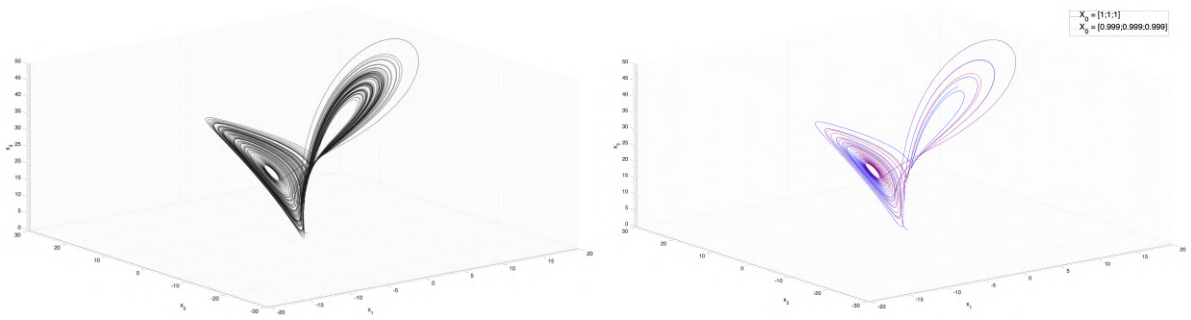
With the advent of quantum computers, public-key cryptosystems are in danger. As an example, Shor [31] showed that using a quantum computer, it is possible to solve the factorization problem and the discrete logarithm problem in a reasonable time. This gave rise to quantum cryptography, which is based on quantum mechanics, and post-quantum cryptography, which is based on mathematical methods resistant to quantum technology, such as certain optimization problems on lattices. In addition to the problems associated with public-key encryption algorithms, there are cryptanalysis methods that exploit weaknesses in the Hardware. For example, for RSA, we find in [16] an acoustic cryptanalysis technique that exploits the sound emitted by RSA decryption in order to reconstruct the secret key. One of the objectives of this thesis is to provide a countermeasure to acoustic cryptanalysis. There are also cryptanalysis techniques that exploit the energy consumed by certain operations of RSA algorithm, such as the rapid exponentiation algorithm, to break the code. These techniques are called Power Analysis techniques, or Side Channel attacks.

2.3 Chaotic cryptography

In this section, I present basic notions on chaos theory and its use in cryptography while highlighting the problem of chaos synchronization. I start by qualitatively defining what a chaotic system is and what it means to synchronize chaos, then I explain how chaos is used in cryptography in current research.

2.3.1 Chaotic Systems.

Chaos theory is the science that studies deterministic dynamical systems having a high sensitivity to initial conditions. A dynamical system $\dot{x} = f(x)$ has high sensitivity to initial conditions if a tiny change in the initial conditions causes a large change in the trajectory of the solution. The sensitivity to initial conditions of chaotic systems is known to the public as the butterfly effect, in reference to the famous conference by meteorologist



(a) The Lorenz strange attractor

(b) The sensitivity to initial conditions

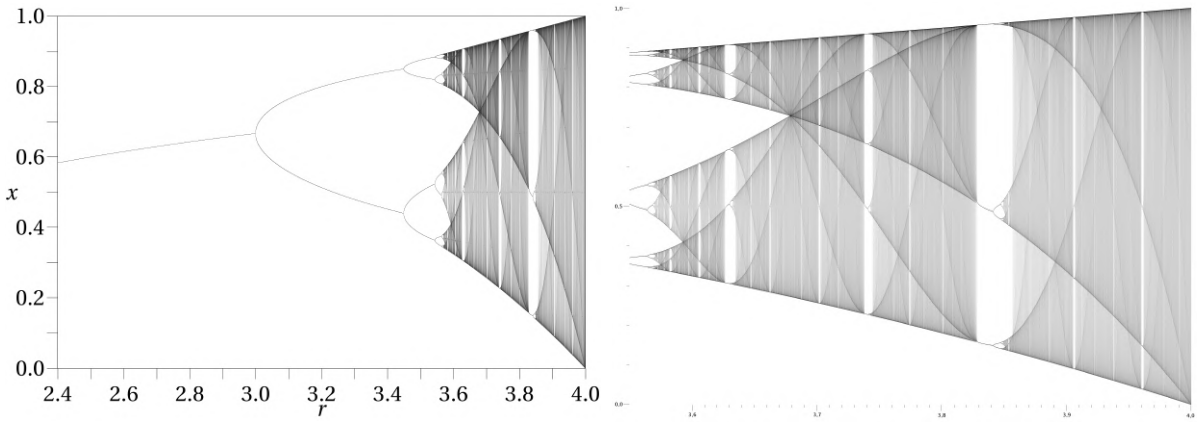
Figure 2.6: The Lorenz system

Edward Lorenz titled: "Predictability: Does the Flap of a Butterfly's Wings in Brazil Set off a Tornado in Texas?" [24]. The objective of this title is to support the fact that in order to know the state of a chaotic system such as the atmosphere, all the details on the initial conditions, however small they may be, are essential. Obviously, a butterfly wing flap generally cannot cause a tornado. The fact remains that its effect is immense, as is the effect of any initial condition on a chaotic system. The only reason why a butterfly wing flap generally cannot cause a tornado is that there is an infinity of other effects which help to "regulate" the atmosphere. Lorenz's simplified model of the atmosphere is given by 2.3.

$$\begin{cases} \dot{x}_1 = \sigma(x_2 - x_1) \\ \dot{x}_2 = x_1(\rho - x_3) - x_2 \\ \dot{x}_3 = x_1 x_2 - \beta x_3 \end{cases} \tag{2.3}$$

Lorenz used : $\sigma = 10$, $\beta = 8/3$ and $\rho = 28$. But for nearby values of these constants, the system will still remain chaotic. For close initial conditions, the trajectories of the system remain close at the beginning and then become different as it is shown in figure 2.6b. During the simulation, we can see that a certain shape appears (figure 2.6a). This shape is called the Lorenz strange attractor, it is the set of values that can take the vector $[x_1, x_2, x_3]^T$. The idea behind this strange attractor is that even if the trajectories are very different for different initial conditions, they still remain in a certain bounded set which is the attractor.

The trajectory of a chaotic system for a given initial condition is called an orbit. For continuous time chaotic systems like the Lorenz system, the orbit is a continuous curve. But there are also discrete-time chaotic systems of the form $x_{n+1} = f(x_n)$ for which the orbit is a set of discrete points. As an example of such systems: The logistic map. It is an



(a) The Bifurcation diagram of the logistic map

(b) The fractal structure of 2.7a

Figure 2.7: The bifurcation diagram

idealized model of population growth given by 2.4.

$$x_{n+1} = rx_n(1 - x_n) \tag{2.4}$$

The parameter that tells us if the system is chaotic or not is the value r. For example, if we take $r = 2$, the system will converge to the value 0.5 for any initial condition, so the system is not chaotic. For $r = 3.1$, the system oscillates between two values: 0.76 and 0.56, it is called a periodic orbit. For $r = 3.57$, we have no longer oscillations of finite periods (because we have oscillations between an infinite number of values). It is the beginning of Chaos. We can do a graph of the different values that can take the system for a large time with respect to r. This graph is called a bifurcation diagram and is shown in figure 2.7a. A bifurcation is a change in the period length of the orbits. We can still see some places beyond $r = 3.57$ where there is no chaos, if we zoom in as in figure 2.7b, we can remark a fractal structure.

For the purpose of my work, I will need sometimes to add an external input m without losing the chaotic behavior of the chaotic system. A system with external input is a system of the form: $\dot{x} = f(x, m)$ where m is the external input. One of them is the modified Rossler's chaotic system which is presented in [12] and is given by 2.5.

$$\begin{cases} \dot{x}_1 = -(x_2 + x_3) \\ \dot{x}_2 = x_1 + ax_2 \\ \dot{x}_3 = b + x_3(x_1 - c) + mx_3 \end{cases} \tag{2.5}$$

For $a = 0.398$, $b = 2$ and $c = 4$, this system is chaotic. The attractor of this system has the form shown in figure 2.8.

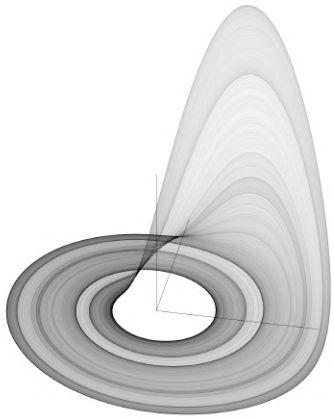


Figure 2.8: The Rossler attractor

2.3.2 The concept of chaos synchronization

Chaotic systems can be implemented in hardware (analog form) or in software (digital form). For the digital case, there is a phenomenon called dynamical degradation, which means that the chaotic behavior of the system will become non-ideal. In this work, I focus exclusively on the analog form of chaotic systems. These systems are designed as electrical circuits. There is however a very important problem: the synchronization of chaotic systems. In chaotic cryptography, we need two versions of the same chaotic signal, one at the emitter and the other at the receiver. We can try to design the same chaotic circuit at the emitter and at the receiver but the parameters will of course never be exactly the same in the two sides of the communication system because of noise and uncertainties. The problem is that chaotic systems are very sensitive to initial conditions, so it is practically impossible to generate the same chaotic signal by this technique. Chaos synchronization is the science that studies how to generate the same chaotic signal as a given chaotic circuit. It really started in 1990 with the works of Pecora and Caroll [26]. The idea is that there is a master chaotic system that will send a signal to another dynamical system (called the slave system) for synchronization so that the slave system generates the chaotic trajectory of the master chaotic system as it is shown in figure 2.9. There are a lot of chaos synchronization techniques. We have for example the chaos synchronization based on state-feedback control, the chaos synchronization based on Backstepping, and the chaos synchronization based on non-linear observers, which is the subject of this thesis. The idea behind the chaos synchronization based on non-linear observers is that at the slave system is an observer that will try to reconstruct the master system's signals.

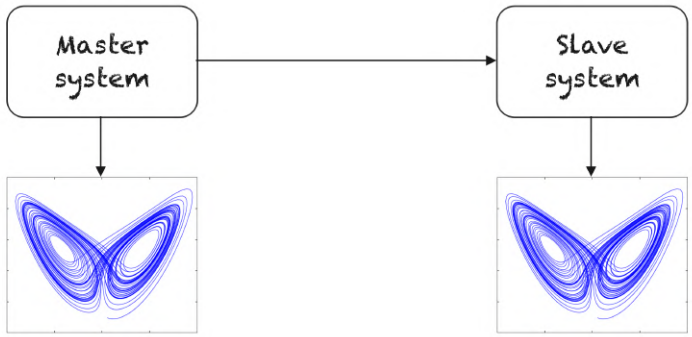


Figure 2.9: The principle of chaos synchronization

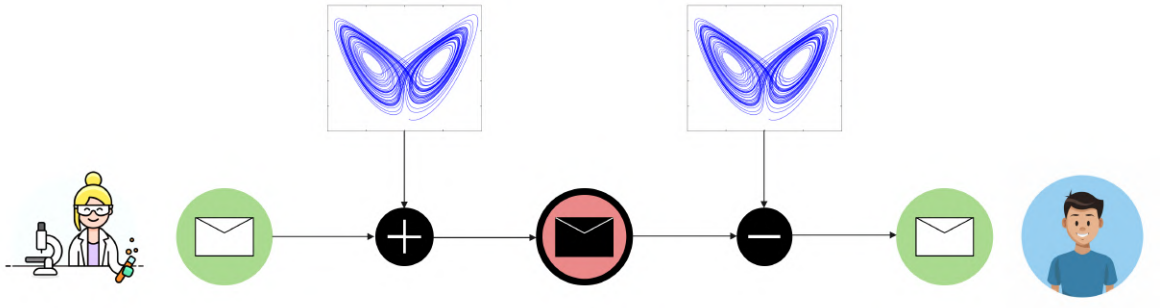


Figure 2.10: Chaotic masking

2.3.3 The design of chaotic cryptosystems

I will present now two designs that are widely used in the literature [8, 17, 21, 36], and which will also be used in this thesis. These designs are chaotic masking and chaotic modulation.

- **Chaotic masking:** The structure of chaotic masking is shown in figure 2.10. The idea is to add to the plaintext the chaotic signal, and then subtract it at the receiver side. It can be done in analog form by adding the chaotic signal to the binary signal, or in the digital form by first converting the chaotic signal into a sequence of bits and then XORing the result with the digital plaintext.
- **Chaotic modulation:** The structure of chaotic modulation is shown in figure 2.11. The idea is to use the plaintext as an external input to the emitter's chaotic system. Then, at the receiver we reconstruct this external input.

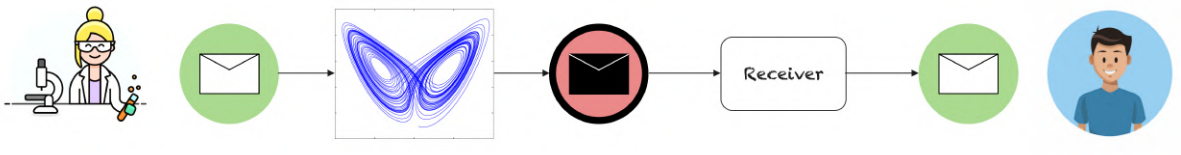


Figure 2.11: Chaotic modulation

2.4 Nonlinear Observers.

As we have seen before, we need to synchronize chaos. In this thesis, this synchronization is achieved using non-linear observers.

The goal of an observer is in general to find the state x (or part of it) of a dynamical system using some knowledge about the output and the input of the system. A dynamical system is described by the general state equation 2.6.

$$\begin{cases} \dot{x} = f(x,m) \\ y = h(x) \end{cases} \tag{2.6}$$

where $x \in \mathbf{R}^n$, $y \in \mathbf{R}^p$ and $m \in \mathbf{R}^q$. The problem of a state observer is to determine x (or part of it) knowing m and y. Sometimes m is not known and the goal of the observer is to estimate x and m using y. Before we can design a state observer, we need to know if it is possible to observe the state vector x. This gives rise to the notion of observability of dynamical systems. The question is the following: Given a dynamical system represented by the functions f and h, and knowing y and u, can we reconstruct x ?.

2.4.1 The observability problem

The notion of observability depends on the output vector y, the input vector m and the structure of the dynamical system (the functions f and h). Let us first analyze the observability with respect to y.

For a given input signal m, let us consider the function 2.7.

$$\begin{aligned} \Phi_m : \mathbf{I} &\to \Gamma \\ x_0 &\mapsto \Phi_m(x_0) = h(\{x_m(t,x_0)\}) \end{aligned} \tag{2.7}$$

where $I \subset R^n$ is the set of all initial conditions of the dynamical system, Γ is the set of all the curves on $\mathbf{R}^p$ and $\{x_m(t,x_0)\}$ is a solution for the input m and given the initial condition x_0[2]. This function allows us to define the observability given a certain input signal m. All the following definitions concerning observability are reformulations I did of the definitions of [5] using the function Φ_m.

Definition 2: *Observability*
The system 2.6, under the application of an input signal m, is said to be observable if Φ_m is injective. In other words, there is no ambiguity in the state trajectory given a certain output and a certain input.

Definition 3: *Weak observability*
The system 2.6, under the application of an input signal m, is said to be weakly observable if for every $x_0 \in \mathbf{I}$ there is a neighborhood V of x_0 such that the restriction of Φ_m to V is injective.

Sometimes, observability will appear only after a certain period of time. It is the case for example of two state curves that have the same output during an interval of time $[t_0, t_f]$ and whose corresponding outputs are not necessarily equal after t_f. This gives rise to the notion of local weak observability:

Definition 4: *Local weak observability*
The system 2.6, under the application of an input signal m, is said to be locally weakly observable if for every $x_0 \in \mathbf{I}$ there is a neighborhood V of x_0 such that for any neighborhood W of x_0 contained in V, the restriction of Φ_m to W is injective when considering time intervals for which trajectories remain in V.

In practice, we have a formula to check if the system is observable or not. This formula is known as the observability rank condition.

Definition 5 [5]: *The observability rank condition*
The system 2.6, under the application of an input signal m, is said to satisfy the observ-

[2]The function Φ_m is well defined iff the Cauchy-Lipschitz theorem holds.

ability rank condition if 2.8 holds.

$$rank\{[\frac{\partial h(x)}{\partial x} \quad \frac{\partial L_f h(x)}{\partial x} \quad \frac{\partial L_f^2 h(x)}{\partial x} \quad ... \quad \frac{\partial L_f^{n-1} h(x)}{\partial x}]^T\} = n \tag{2.8}$$

where 2.9,

$$L_f^i h(x) = L_f(L_f^{i-1} h(x)) \tag{2.9}$$

and $L_f h(x)$ is the Lie derivative of h along f defined by 2.10.

$$L_f h(x) = \frac{\partial h(x)}{\partial x} f(x,m) \tag{2.10}$$

For linear time invariant (LTI) systems ($f(x,m) = Ax + Bm$ and $h(x) = Cx$), we have 2.11.

$$h(x) = Cx \Rightarrow \frac{\partial h(x)}{\partial x} = C \Rightarrow \frac{\partial L_f h(x)}{\partial x} = CA \Rightarrow ... \Rightarrow \frac{\partial L_f^{n-1} h(x)}{\partial x} = CA^{n-1} \tag{2.11}$$

which gives us the observability rank condition for LTI systems 2.12.

$$rank\{[C \quad CA \quad CA^2 \quad ... \quad CA^{n-1}]^T\} = n \tag{2.12}$$

The relation between observability and the observability rank condition is given by the following theorem:

Theorem [5]:
If the system 2.6, under the application of an input signal m, satisfies the observability rank condition, then it is locally weakly observable. Conversely, if the system 2.6 is locally weakly observable, then it satisfies the observability rank condition in an open dense subset of $\mathbf{R}^n$.

We will now focus our attention on the observability with respect to the input vector m. For a certain input signal m, we can study the observability of the system with respect to the output. This observability can of course be affected by the input, which gives rise to the following definition:

Definition 6 [5]: *Uniformly observable systems*
A system is uniformly observable (UO) if it is observable for any input signal m. If the system is (UO) only in an interval of time $[0,t]$, we say that it is locally uniformly observable.

For a system that is (UO), the observability does not depend on the input.

2.4.2 Classical observers

Once we know if the dynamical system is observable, we can design observers. I will start with classical observers which are of two types: Linear observers and non-linear observers. Then, I will describe the more advanced observers.

2.4.2.1 Linear Observers

• *Kalman filter for stochastic LTI systems :*

Let us consider the stochastic LTI system given by 2.13.

$$\begin{cases} \dot{x} = Ax + Bm + w \\ y = Cx + v \end{cases} \tag{2.13}$$

where w is the process noise and v is the measurement noise (these noises are assumed to be Gaussian). If 2.13 is observable, then there exists an observer of the form 2.14.

$$\dot{\hat{x}} = A\hat{x} + Bm - L(C\hat{x} - y) \tag{2.14}$$

where $L = PM^T R^{-1}$, such that P satisfies the Algebraic Riccati equation (ARE) given by 2.15.

$$AP + PA^T + Q - PM^T R^{-1} MP = 0 \tag{2.15}$$

where R and Q are the covariance matrices of v and w respectively.

• *Luenberger observer for deterministic LTI systems:*

Let us consider the deterministic LTI system given by 2.16.

$$\begin{cases} \dot{x} = Ax + Bm \\ y = Cx \end{cases} \tag{2.16}$$

If 2.16 is observable, then there exists an observer of the form 2.17.

$$\dot{\hat{x}} = A\hat{x} + Bm - L(C\hat{x} - y) \tag{2.17}$$

where L is such that $(A - LC)$ is stable.

2.4.2.2 Non-linear Observers

• *Extended Kalman filter:*

In this method we linearize 2.6 in the neighborhood of our operating point, and then we apply the Kalman filter by considering the non-linearities as noises.

• *Thau's observer:*

Let us consider the system given by 2.18.

$$\begin{cases} \dot{x} = Ax + g(t,m,y) + f(x) \\ y = Cx \end{cases} \tag{2.18}$$

The Thau's observer for 2.18 is given by 2.19.

$$\dot{\hat{x}} = A\hat{x} + g(t,m,y) + f(\hat{x}) - L(C\hat{x} - y) \tag{2.19}$$

This observer converges if 2.18 is observable, f is globally Lipschitz with a Lipschitz constant γ and if L satisfies an equation of the form 2.20.

$$(A - LC)^T P + P(A - LC) = -Q \tag{2.20}$$

where P and Q are positive definite matrices that satisfy the inequality 2.21.

$$\gamma < \frac{\lambda_{min}(Q)}{2\lambda_{max}(P)} \tag{2.21}$$

• *High gain observer (Ragahvan's method):*

The High gain observer is for systems of the form 2.18. Let us assume that the system is observable and that f is globally Lipschitz with a Lipschitz constant γ. The high gain observer has the same form as Thau's observer except that the determination of L is different. To determine L, we use the Ragahvan's method:

1- Set $\epsilon > 0$.
2- Solve the Riccati equation for P given by 2.22.

$$AP + PA^T + P(\gamma^2 I - \frac{C^T C}{\epsilon})P + I(\epsilon + 1) = 0 \tag{2.22}$$

3- Check if P is symmetric and positive definite
(i) If yes, set $L = \frac{PC^T}{2\epsilon}$.
(ii) If no, set $\epsilon = \frac{\epsilon}{2}$ and repeat.

2.4.3 Sliding mode observers

Sliding mode observers are very efficient for linear systems with unknown uncertainties. These systems are given by 2.23.

$$\begin{cases} \dot{x} = Ax + Bm + Df(x,m,t) \\ y = Cx \end{cases} \tag{2.23}$$

where D is a matrix of the appropriate dimension and f is the unknown uncertainty.

Let us assume that f is bounded by some scalar ρ, i.e $||f(x,m,t)|| \leq \rho, \quad \forall x \in \mathbf{R}^n, \forall m \in \mathbf{R}^q, \forall t \geq 0$. There are three types of sliding mode observers which are widely used in practice: The Walcott-Zak sliding mode observer, the Edwards-Spurgeon sliding mode observer and the Higher order sliding mode observers. I will present now each one of them.

2.4.3.1 The Walcott-Zak sliding mode observer

For the Walcott-Zak sliding mode observer [34], we add a structural condition on the unknown uncertainty. Namely, we assume that there are two positive definite matrices P and Q and two matrices of appropriate dimensions L and F such that 2.24.

$$\begin{cases} (A-LC)^T P + P(A-LC) = -Q \\ PD = C^T F^T \end{cases} \tag{2.24}$$

We also assume that the pair (A,C) is observable. The Walcott-Zak observer is given by 2.25.

$$\dot{\hat{x}} = A\hat{x} + Bm - L(C\hat{x} - y) + \mu \tag{2.25}$$

where μ is a discontinuous function defined by 2.26.

$$\mu = \begin{cases} -\rho \dfrac{P^{-1}C^T F^T FCe}{||FCe||}, & if \quad FCe \neq 0 \\ 0, & if \quad FCe = 0 \end{cases} \tag{2.26}$$

For details about the convergence of this observer, you can refer to [33]. The principal disadvantage of this technique is that the discontinuity of μ gives rise to high-frequency oscillations. The apparition of these high-frequency oscillations is called the chattering effect.

2.4.3.2 The Edwards-Spurgeon sliding mode observer

For Edwards-Spurgeon sliding mode observer [13], we define the sliding surface $S := \{e \in \mathbf{R}^n : Ce = 0\}$ where $e = x - \hat{x}$. Let us assume that $rank(CD) = rank(D)$ and that the invariant zeroes of (A, D, C) are in $\mathbf{C}_-$. It was proved in [13] that under these assumptions there exist a non-singular change of coordinates $x \mapsto Tx$ that transforms 2.23 into 2.27.

$$\begin{cases} \dot{x}_1 = A_{11}x_1 + A_{12}x_2 + B_1 m \\ \dot{x}_2 = A_{21}x_1 + A_{22}x_2 + B_2 m + D_2 f(x, m, t) \\ y = x_2 \end{cases} \tag{2.27}$$

where $x_1 \in \mathbf{R}^{n-p}$, $x_2 \in \mathbf{R}^p$ and A_{11} is Hurwitz.

The Edwards-Spurgeon sliding mode observer is given by 2.28.

$$\begin{aligned} \dot{\hat{x}}_1 &= A_{11}\hat{x}_1 + A_{12}\hat{x}_2 + B_1 m \\ \dot{\hat{x}}_2 &= A_{21}\hat{x}_1 + A_{22}\hat{x}_2 + B_2 m - (A_{22} - A_{22}^s)e_y + \nu \\ \hat{y} &= \hat{x}_2 \end{aligned} \tag{2.28}$$

where A_{22} is Hurwitz, $e_y = y - \hat{y}$ and ν is a discontinuous function given by 2.29.

$$\nu = \begin{cases} -\rho ||D_2|| \dfrac{P_2 e_y}{||P_2 e_y||}, & if \quad e_y \neq 0 \\ 0, & if \quad e_y = 0 \end{cases} \tag{2.29}$$

where P_2 is a Lyapunov matrix for A_{22}^s. The state estimate is given by 2.30.

$$\hat{x} = T^{-1}[\hat{x}_1, \hat{x}_2]^T \tag{2.30}$$

One problem with Edwards-Spurgeon sliding mode observer is that the hypotheses are not always verified.

2.4.3.3 Higher-order sliding mode observers

Higher-order sliding mode observers [3, 15, 22, 23] are for dynamical system in the triangular form 2.31.

$$\begin{cases} \dot{x} = A_n x + H_n V_n(x, w) \\ y = C_n x \end{cases} \tag{2.31}$$

where $w \in \mathbf{R}$ is an unknown input and

$$A_n = \begin{bmatrix} 0 & 1 & ... & 0 \\ 0 & 0 & 1 & ... \\ ... & ... & ... & 1 \\ 0 & ... & ... & 0 \end{bmatrix}, \; H_n = \begin{bmatrix} 0 & ... & 1 \end{bmatrix}^T, \; C_n = \begin{bmatrix} 1 & 0 & ... & 0 \end{bmatrix}$$

Let us assume that the state is uniformly bounded, i.e $\exists (d_1, ..., d_n) \in \mathbf{R}^n, \; s.t \; \forall t > 0, \forall i \in \{1, 2, ..., n\} : \; |x_i(t)| < d_i$. Let us also assume that w and its derivative are bounded.

There are several types of Higher-order sliding mode observers. For example, the Higher-order sliding mode observer presented in [3], which has the form 2.32 is a second order sliding mode observer.

$$\begin{aligned} \dot{\hat{x}}_1 &= z_1 + \lambda_1 \sqrt{|x_1 - \hat{x}_1|} sign(x_1 - \hat{x}_1) \\ \dot{z}_1 &= \alpha_1 sign(x_1 - \hat{x}_1) \\ \dot{\hat{x}}_2 &= z_2 + \lambda_2 \sqrt{|z_1 - \hat{x}_2|} sign(z_1 - \hat{x}_2) \\ \dot{z}_2 &= \alpha_2 sign(z_1 - \hat{x}_2) \\ &... \\ \dot{\hat{x}}_{n-1} &= z_{n-1} + \lambda_{n-1} \sqrt{|z_{n-2} - \hat{x}_{n-1}|} sign(z_{n-2} - \hat{x}_{n-1}) \\ \dot{z}_{n-1} &= \alpha_{n-1} sign(z_{n-2} - \hat{x}_{n-1}) \\ \dot{\hat{x}}_n &= z_n + \lambda_n \sqrt{|z_{n-1} - \hat{x}_n|} sign(z_{n-1} - \hat{x}_n) \\ \dot{z}_n &= \alpha_n sign(z_{n-1} - \hat{x}_n) \end{aligned} \tag{2.32}$$

where λ_i and α_i are the observer gains. They are positive scalars that we need to define. There are other higher order sliding mode observers like third order sliding mode observers [11] and fourth order sliding mode observers [29]. The main advantage of Higher order sliding mode observers over classical sliding mode observers is that the chattering effect is reduced. It does not mean that there is no chattering effect in Higher order sliding mode observers, it is just reduced. If we want to eliminate it almost completely we can replace the discontinuous functions as the sign function with fuzzy inference systems [27, 28].

2.4.4 Unknown inputs observers

The goal of an unknown inputs observer is to estimate the state of a dynamical system without having knowledge about part of (or all) the input. There are a lot of techniques of

design of unknown inputs observers. One of them is to separate the state vector into two parts: one part which is influenced by the unknown inputs and the other which is not. It is possible to design with this method a reduced order observer [35]. Other methods are called algebraic methods of design and they are based on the resolution of matrix linear equations [10]. In the algebraic methods of design, we consider linear systems of the form 2.33.

$$\begin{cases} \dot{x} = Ax + Bu + Fw \\ y = Cx \end{cases} \tag{2.33}$$

where u is the known input vector and w the unknown input vector. Let us assume that F is full rank and that (A, C) is observable. The full order unknown input observer presented in [6, 10] is given by 2.34.

$$\begin{aligned} \dot{z} &= Nz + Gu + Ly \\ \hat{x} &= z - Ey \end{aligned} \tag{2.34}$$

where N, G, L and E are matrices that we need to determine such that the observer converges. The dynamics of the observer error $e = x - \hat{x}$ is given by 2.35.

$$\dot{e} = Ne + (PB - G)u + (PA - NP - LC)x \tag{2.35}$$

where $P = I + EC$. The error e converges asymptotically to zero if and only if N is stable, $P = I + EC$, $LC = PA - NP$, $G = PB$ and $PF = 0$. It is a set of matrix linear equations. The sufficient and necessary conditions for the existence of a solutions to these equations are given by 2.36.

$$\begin{aligned} &(i)\ \ rank(CF) = rank(F) \\ &(ii)\ \ rank \begin{bmatrix} sP - PA \\ C \end{bmatrix} = n, \quad \forall s \in \mathbf{C}, \quad Re(s) \geq 0 \end{aligned} \tag{2.36}$$

In [9], this result is extended to the case where the unknown inputs affect the state and affect also the output as in 2.37.

$$\begin{cases} \dot{x} = Ax + Bu + F_1 w \\ y = Cx + F_2 w \end{cases} \tag{2.37}$$

Let us assume that 2.37 satisfies a structural constraint of the form 2.38.

$$\begin{aligned} rank \begin{bmatrix} CF_1 & F_2 \\ F_2 & 0 \end{bmatrix} &= rank(G) + rank \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} \\ rank \begin{bmatrix} sI - A & -F_1 \\ C & F_2 \end{bmatrix} &= n + rank \begin{bmatrix} F_1 \\ F_2 \end{bmatrix}, \quad \forall s \in \mathbf{C}, \quad Re(s) \geq 0 \end{aligned} \tag{2.38}$$

The unknown inputs observer presented in [9] for this system is given by 2.39.

$$\begin{aligned} \dot{z} &= Nz + Hu + Jy \\ \hat{x} &= z - Ey \end{aligned} \tag{2.39}$$

The observer error converges asymptotically to zero if and only if N is stable, $P = I + EC$, $PA - NP - JC = 0$, $PF_1 - NEC - JC = 0$, $EF_2 = 0$ and $H = PB$.

The last category of unknown inputs systems that we consider are the singular systems with unknown inputs given by 2.40.

$$\begin{cases} E\dot{x} = Ax + Bu + Fw \\ y = Cx \end{cases} \tag{2.40}$$

The idea is to consider an augmented state vector $\bar{x} = \begin{bmatrix} x \\ w \end{bmatrix}$ and the corresponding augmented system 2.41.

$$\begin{cases} \bar{E}\dot{\bar{x}} = \bar{A}\bar{x} + \bar{B}u \\ y = \bar{C}\bar{x} \end{cases} \tag{2.41}$$

where

$$\bar{E} = \begin{bmatrix} E & 0 \\ 0 & I \end{bmatrix}, \quad \bar{A} = \begin{bmatrix} A & N \\ 0 & 0 \end{bmatrix}, \bar{B} = \begin{bmatrix} B \\ 0 \end{bmatrix}, \quad \bar{C} = \begin{bmatrix} C & 0 \end{bmatrix}$$

An unknown inputs observer for 2.41 is given in [20, 32] by 2.42.

$$\begin{aligned} \dot{z} &= Rz + Hu + Ly \\ \hat{x} &= Mz - Ny \end{aligned} \tag{2.42}$$

If 2.41 satisfies the structural conditions 2.43.

$$\begin{aligned} rank \begin{bmatrix} s\bar{E} - \bar{A} \\ \bar{C} \end{bmatrix} &= n, \quad \forall s \in \mathbf{C} \\ rank \begin{bmatrix} sI - R \\ M \end{bmatrix} &= n, \quad \forall s \in \mathbf{C} \end{aligned} \tag{2.43}$$

then there exists a matrix K of appropriate dimension such that $Kx - \hat{x} \longrightarrow 0$ as $t \longrightarrow +\infty$, $\forall x_0, z_0$ if and only if there exists a matrix P such that R is stable, $P\bar{A} - RP\bar{E} - L\bar{C} = 0$, $K = MP\bar{E} + N\bar{C}$ and $H = P\bar{B}$.

2.4.5 Adaptive observers

The goal of adaptive observers is to estimate the state vector and parameters of the system. Let us consider the class of systems with unknown parameters given by 2.44.

$$\begin{cases} \dot{x} = f(x,u,t) + g(x,u,t)\theta \\ y = h(x) \end{cases} \tag{2.44}$$

where $\theta \in \mathbf{R}^q$ is the unknown parameters vector. An adaptive observer for 2.44 which had been proposed in [4]. It is given by 2.45.

$$\begin{aligned} &\dot{\hat{x}} = f(y,\hat{z}u,t) + g(y,\hat{z},u,t)\hat{\theta} + k(h(\hat{x}-y,t) \\ &\hat{x} = [\hat{y},\hat{z}]^T \end{aligned} \tag{2.45}$$

where $\hat{\theta}$ is updated using the adaptation law 2.46.

$$\dot{\hat{\theta}} = -\Lambda\phi^T(\hat{y}-y,y,\hat{z},u,t) \tag{2.46}$$

where $\Lambda = \Lambda^T > 0$. To ensure the convergence of $\hat{\theta}$, the function g must satisfy the persistent excitation condition [4]: $\exists T, k_1, k_2 > 0$ s.t $\forall t \geq 0$ we have 2.47.

$$k_1 I_q \geq \int_t^{t+T} g(y(\tau),\hat{z}(\tau),u(\tau),\tau)g^T(y(\tau),\hat{z}(\tau),u(\tau),\tau)d\tau \geq k_2 I_q \tag{2.47}$$

The adaptive observer 2.45 for the system 2.44, with $\dot{\theta} = 0$, converges asymptotically if there exists a decreasing positive definite function $V(t,e)$, with $e = \hat{x} - x = [\hat{y}-y, \hat{z}-z]^T = [e_y, e_z]^T$, of class $\mathscr{C}^1$, with $|(\frac{\partial V}{\partial e})(t,e)|$ a decreasing function, and a continuous function $k(e_y,t)$ bounded with respect to t with $k(0,t) = 0$, such that $\forall u$, $\forall e$, $\forall y$, $\forall \sigma$, $\forall \alpha > 0$, $\forall t \geq 0$, we have 2.48.

$$\begin{aligned} &(i)\ \ \dot{V} + \frac{\partial V}{\partial e}[f(y,\sigma,u,t) - f(y,\sigma - e_z,u,t) + \\ &(g(y,\sigma,u,t) - g(y,\sigma-e_z,u,t))\theta + k(e_y,t)] \leq -\alpha|e|^2 \\ &(ii)\ \ \frac{\partial V}{\partial e}g(y,\sigma,u,t) = \phi(e_y,y,\sigma,u,t) \end{aligned} \tag{2.48}$$

and g is globally bounded and f, g are globally Lipschitz with respect to z, uniformly with respect to (u,y,t).
If in addition to this, the function g satisfies the persistant excitation condition and $\dot{g}$ is bounded, then $||\hat{\theta} - \theta|| \longrightarrow 0$ as $t \longrightarrow +\infty$.

A special case of 2.44 is given by 2.49 [7].

$$\begin{cases} \dot{x} = Ax + \psi_1(u,x) + B\psi_2(u,x)\theta \\ y = Cx \end{cases} \tag{2.49}$$

where ψ_1 and ψ_2 are assumed to be globally Lipshitz with Lipshitz constants k_1 and k_2 respectively. We also assume that 2.49 is of minimum phase and that there exists two positive definite matrices P, Q and a matrix L such that 2.50.

$$\begin{aligned} &P(A-LC)+(A-LC)^T P = -Q \\ &PB = C^T \\ &k_1 + k_2 max(\theta)|B| < \frac{\lambda_{min}(Q)}{2\lambda_{max}(P)} \end{aligned} \tag{2.50}$$

It is possible to construct under these conditions an adaptive observer [7].

We can find generalization of adaptive observers for linear MIMO systems in [39]. Other generalizations have been done for non-linear uniformly observable SO (single output) systems [37] and non-linear uniformly observable MIMO systems with non-linear parametrization [14].

2.4.6 ANFIS (Adaptive Neuro-Fuzzy Inference Systems) observers

An ANFIS observer is a neural network whose training optimizes a fuzzy system. For state estimation using ANFIS, one idea would be to train an ANFIS to predict the states given certain information as the output and the input of the dynamical system. We can also use ANFIS to predict the unknown input of a dynamical system given its state vector (or its output vector or both). In all these cases the principle is the same: we train a neural network to predict some signal given a set of other signals. ANFIS can be used in multiple ways. In this thesis, I will use ANFIS for the cryptanalysis of chaos modulation cryptosystems, by estimating the unknown input using the information of the output, and I will also use it to generate white noise given some chaotic signal as an input.

2.5 Conclusion

In this chapter, I started with a brief overview of classical cryptography. Then, I recalled the basics of chaotic systems and how they could be used in cryptography. I also exposed the problem of synchronization of chaotic systems which is of fundamental importance in chaotic cryptography. The approach used in this thesis for chaos synchronization is the use of non-linear observers. I have therefore recalled the theory of non-linear observers. First, I introduced the notion of observability for a dynamical system, then I presented

how to design observers. The observers that have been presented are first of all the classical observers, whether linear or non-linear, then the more advanced observers: Sliding mode observers, Unknown inputs observers, Adaptive observers, and ANFIS observers.

CHAPTER 3

CHAOS SYNCHRONIZATION USING ADAPTIVE UNKNOWN INPUTS OBSERVERS AND ADAPTIVE SLIDING MODE UNKNOWN INPUTS OBSERVERS

3.1 Introduction

The objective of this chapter is to present two non-linear observers for the synchronization of chaotic systems: an adaptive unknown inputs observer, constructed in [12] and an adaptive sliding mode unknown inputs observer also constructed in [12]. I will try to improve the performance of these observers and a comparison will be made between the original observers and the improved versions of these. I will also build a chaotic stream cipher and a chaotic cryptosystem for the transmission of audio signals within which the chaotic synchronization is done using the observers presented in this chapter. I will also present a technique to improve the frequency characteristics of chaotic systems so that the chaotic cryptosystems become more robust to frequency attacks.

3.2 Adaptive unknown inputs observer

I present in this section an adaptive unknown inputs observer which has been constructed in [12]. Its main interest is the robust estimation of unknown inputs of chaotic systems. The inputs that I want to estimate in this chapter are binary signals. As we will see in the following paragraphs, the estimation presents a convergence time at each change

of bit, which can be problematic if the reconstructed binary signal is not an estimate of a plaintext, but an estimate of a secret sequence used for the encryption of a plaintext. I will try to remedy this by adding certain equations to the observer[1]. I will end this section with the design of a stream cipher and I will present a technique to improve the frequency characteristics of chaotic systems.

3.2.1 The original observer

The unknown inputs observer constructed in [12] is for systems of the form 3.1.

$$\begin{cases} \dot{x} = Ax + Bf(x) + Bg(x)m(t) + Fd(t) \\ y = Cx + Gd(t) \end{cases} \tag{3.1}$$

where f and g are $\mathscr{C}^1$ functions, A, B, C, F and G are constant matrices of appropriate dimensions, $m(t)$ is the vector of unknown inputs and $d(t)$ is a perturbation.

Let us assume that $m(t)$ is a piecewise continuous function whose supremum is bounded by some constant $K_m > 0$ and whose derivative is almost always zero, and that d is Lebesgue measurable. Let us also assume that the solutions $x(t)$ of 3.1 are globally, uniformly bounded, which is the case if 3.1 is a chaotic system.

The Adaptive unknown inputs observer for 3.1 is given by 3.2.

$$\begin{cases} \dot{z} = Nz + Jy + Hf(\hat{x}) + Hg(\hat{x})\hat{m} + \frac{1}{2}\hat{\beta}HM(Ty - C_1\hat{x}) \\ \hat{x} = z - Ey \end{cases} \tag{3.2}$$

with the adaptation laws 3.3.

$$\begin{cases} \dot{\hat{m}} = \delta g(\hat{x})^T M(Ty - C_1\hat{x}) \\ \dot{\hat{\beta}} = \gamma |M(Ty - C_1\hat{x})|^2 \end{cases} \tag{3.3}$$

[1]The change I made can be used for all estimators of binary unknown inputs. It is a secondary estimation layer that makes it possible to pass from a continuous estimate, solution of differential equations, to a truly binary signal without a convergence time at each bit change.

where $\delta, \gamma \in \mathbf{R}^+$, and N, E, J, H, T, M, C_1 are constant matrices of appropriate dimensions that satisfy 3.4.

$$\begin{aligned} EG &= 0 \\ PA - NP - JC &= 0 \\ PF - NEG - JG &= 0 \\ H &= PB \\ P &= I + EC \\ TG &= 0 \\ TC &= C_1 \end{aligned} \tag{3.4}$$

and (N, H, C_1) satisfies conditions of the form 3.5.

$$\begin{aligned} N^T Q + QN &< 0 \\ H^T Q &= MC_1 \end{aligned} \tag{3.5}$$

Under these assumptions, the estimated state $\hat{x}$ converges asymptotically to the real state x [12]. To ensure the parametric convergence of $\hat{m}$ and $\hat{\beta}$, the function g has to be persistently exciting, i.e. $\exists \mu, T_0 > 0$ such that 3.6 holds.

$$\int_t^{t+T_0} g(x(s))^T H^T H g(x(s))\, ds \geq \mu, \quad \forall t \geq 0 \tag{3.6}$$

The design of the observer is as follows:

Step 1 : compute C_1 using the formula 3.7.

$$C_1 = (I - GG^+)C \tag{3.7}$$

Step 2 : compute T using the formulas 3.8 and 3.9.

$$X = R_2 R_1^+ - Z(I - R_1 R_1^+) \tag{3.8}$$

$$X = T, \;\; R_1 = [C \; G], \;\; R_2 = [C_1 \; 0] \tag{3.9}$$

Step 3 : if $p = n$, choose $E = T$. Else, if $p < n$, choose $E = [T \; 0]^T$.

Step 4 : compute H and P using the formulas 3.10 and 3.11.

$$H = PB \tag{3.10}$$

$$P = I + EC \tag{3.11}$$

Step 5 : compute A_1 using the formula 3.12.

$$A_1 = PA - PFG^+C \tag{3.12}$$

Step 6 : find the matrices M, L and Q by solving the convex optimization problem 3.13, 3.13, 3.14, 3.15.

$$Min \ \ \rho,$$

$$Q > 0 \tag{3.13}$$

$$QA_1 + A_1^T Q + WC_1 + C_1^T W^T < 0 \tag{3.14}$$

$$\begin{bmatrix} \rho I & H^T Q - MC_1 \\ QH - C_1^T M^T & \rho I \end{bmatrix} \leq 0 \tag{3.15}$$

$$L = -Q^{-1}W \tag{3.16}$$

Step 7 : compute K and N using the formulas 3.17 and 3.18.

$$K = -PFG^+ - L(I - GG^+) \tag{3.17}$$

$$N = A_1 - LC_1 \tag{3.18}$$

Step 8 : compute J using the formula 3.19.

$$J = -NE - K \tag{3.19}$$

3.2.2 Example : The perturbed Rossler system with external input

One example of a system of the form 3.1 is the perturbed Rossler chaotic system with external input 3.20 [12].

$$\begin{cases} \dot{x}_1 = -(x_2 + x_3) + d \\ \dot{x}_2 = x_1 + ax_2 + d \\ \dot{x}_3 = b + x_3(x_1 - c) + mx_3 + d \\ y_1 = x_1 + 2d \\ y_2 = x_3 + d \end{cases} \tag{3.20}$$

or in the matrix form 3.21.

$$\begin{cases} \dot{x} = \begin{bmatrix} 0 & -1 & -1 \\ 1 & a & 0 \\ 0 & 0 & -c \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} (b + x_1x_3) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} x_3 m(t) + \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} d(t) \\ y = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} x + \begin{bmatrix} 2 \\ 1 \end{bmatrix} d(t) \end{cases} \tag{3.21}$$

The matrices of the observer are given by [12]:

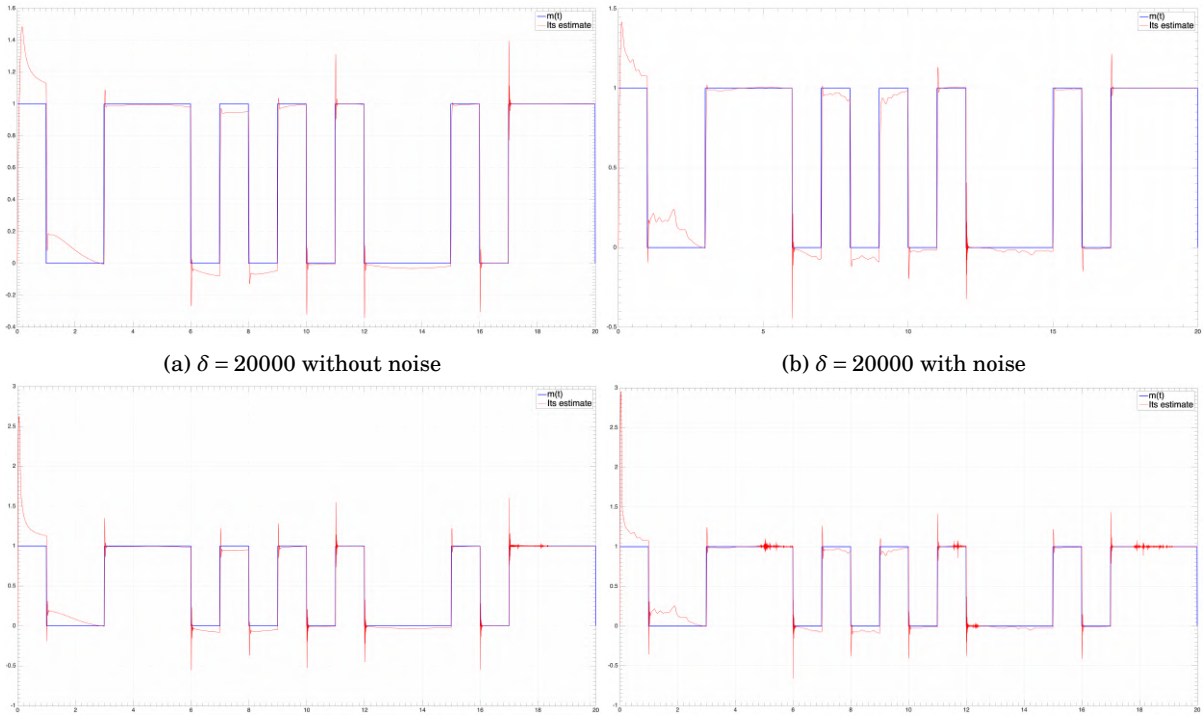
$$T = \begin{bmatrix} 0.2 & -0.4 \\ -0.4 & 0.8 \end{bmatrix}, \; E = \begin{bmatrix} 0.2 & -0.4 \\ -0.4 & 0.8 \\ 0 & 0 \end{bmatrix}, \; H = \begin{bmatrix} -0.4 \\ 0.8 \\ 1 \end{bmatrix}, \; P = \begin{bmatrix} 1.2 & 0 & -0.4 \\ -0.4 & 1 & 0.8 \\ 0 & 0 & 1 \end{bmatrix},$$

$$A_1 = \begin{bmatrix} -0.32 & -1.2 & 0.24 \\ 0.44 & 0.798 & -3.08 \\ -0.4 & 0 & -4.2 \end{bmatrix}, \; C_1 = \begin{bmatrix} 0.2 & 0 & -0.4 \\ -0.4 & 0 & 0.8 \end{bmatrix}, \; Q = \begin{bmatrix} 5.1283 & 4.9395 & -2.6505 \\ 4.9395 & 6.5315 & -3.2496 \\ -2.6505 & -3.2496 & 3.0395 \end{bmatrix},$$

$$L = \begin{bmatrix} 19.9972 & -39.9943 \\ -40.0010 & 80.0020 \\ -50.0030 & 100.0059 \end{bmatrix}, \; M^T = \begin{bmatrix} -0.75 \\ 1.5 \end{bmatrix}, \; K = \begin{bmatrix} -20.3172 & 39.8343 \\ 39.441 & -80.282 \\ 49.603 & -100.2059 \end{bmatrix},$$

$$N = \begin{bmatrix} -20.3172 & -1.2 & 40.2343 \\ 40.4410 & 0.798 & -83.082 \\ 49.603 & 0 & -104.2059 \end{bmatrix}, \; J = \begin{bmatrix} 23.9006 & -47.0012 \\ -47.21 & 95.82 \\ -59.5235 & 120.0471 \end{bmatrix}$$

I did simulations on MATLAB/SIMULINK (with "auto" time step) for this observer, for given inputs and adaptation gains. In all the simulations I fix $\gamma = 5000$ and I change δ. The results are shown in figure 3.1.



(a) $\delta = 20000$ without noise

(b) $\delta = 20000$ with noise

(c) $\delta = 80000$ without noise

(d) $\delta = 80000$ with noise

Figure 3.1: The simulations of the original observer

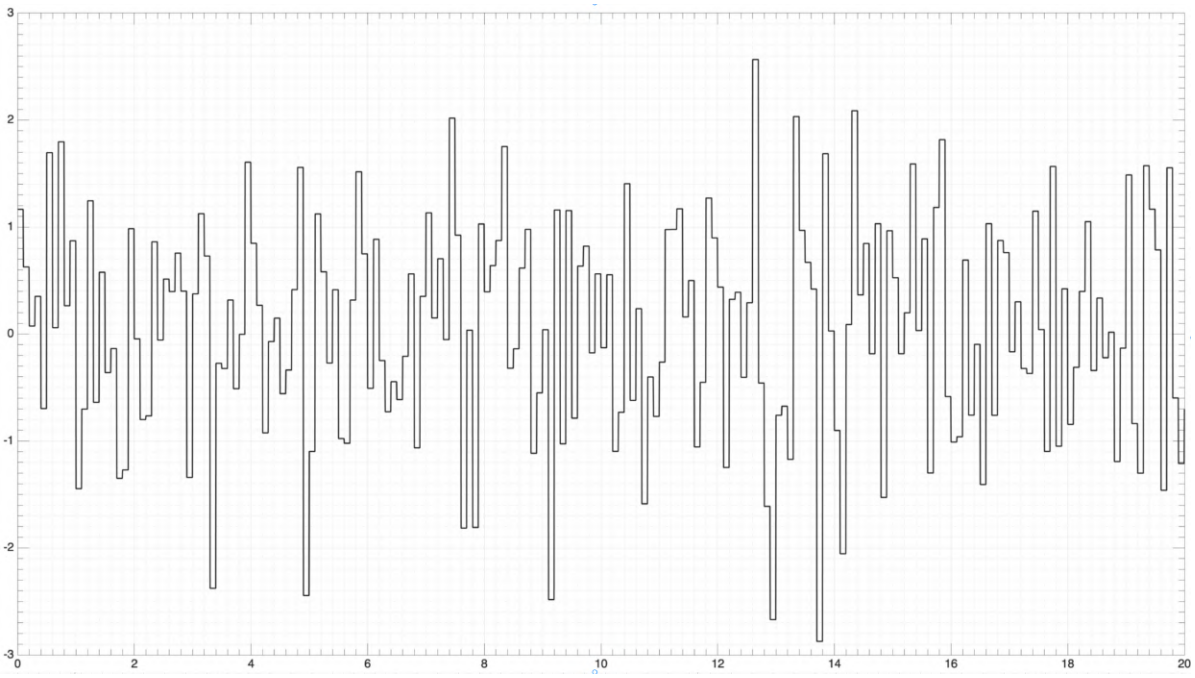
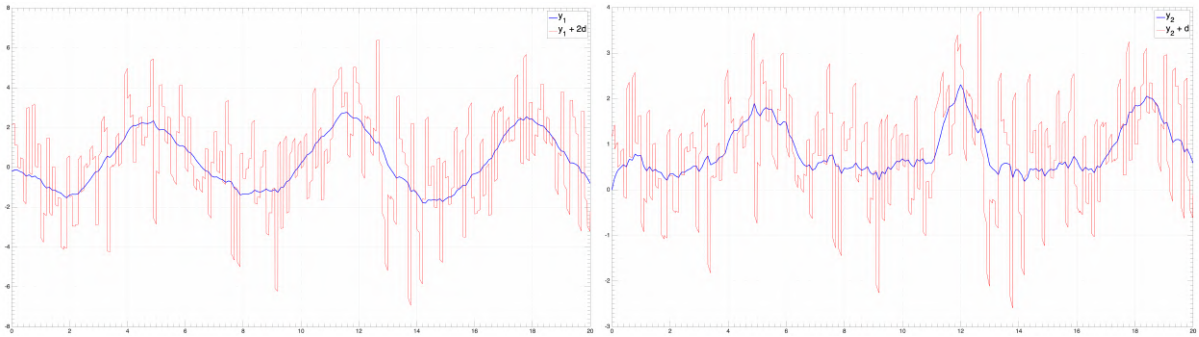


Figure 3.2: The noise d



(a) The noisy first output $y_1 + 2d$

(b) The noisy second output $y_2 + d$

Figure 3.3: The noisy outputs

When we compare figure 3.1a to figure 3.1c, we remark that when δ grows up, the oscillations grow up too. It is due to the fact that the slope of $\hat{m}$ is proportional to δ. For figure 3.1b and figure 3.1d, the noise that I added is shown in figure 3.2. We can see in figure 3.3a and figure 3.3b the noisy versions of the outputs. As we can see, the observer is robust to noise.

We can use a low pass filter 3.22 to avoid the high frequency oscillations. The result of such filtering on the estimate in figure 3.1d is shown in figure 3.4, where $f = 4.9Hz$ and

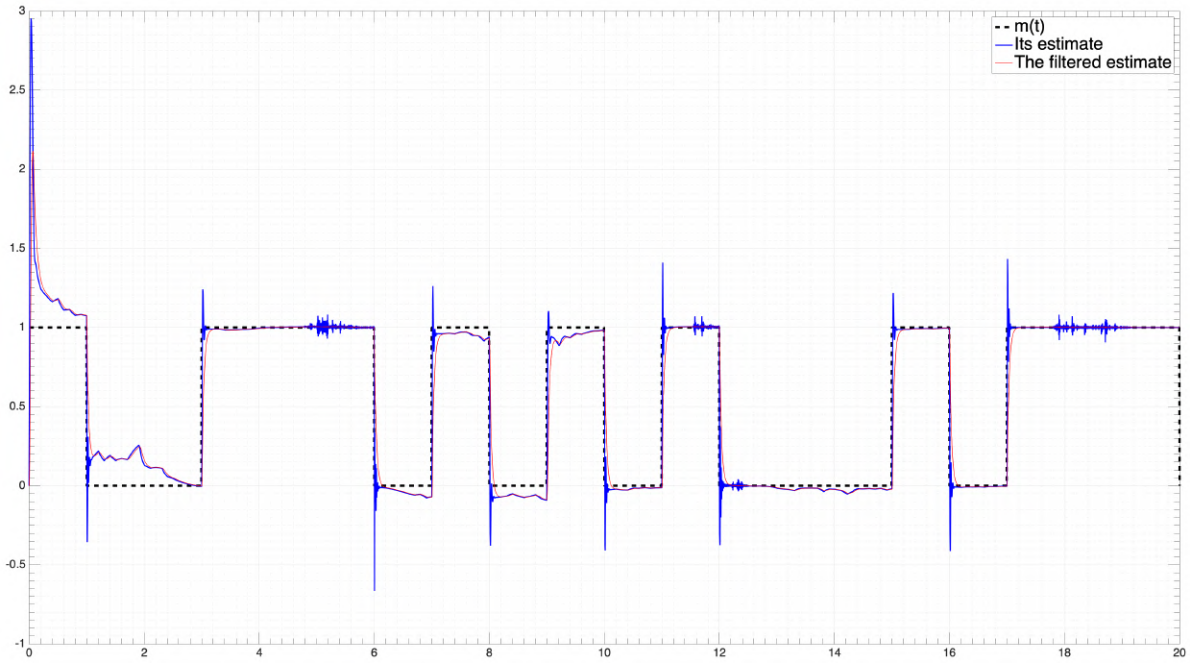


Figure 3.4: A filtered version of 3.1d

the initial condition of the filter is set to 0.

$$H(s) = \frac{2\pi f}{s + 2\pi f} \tag{3.22}$$

3.2.3 The modified adaptive unknown inputs observer

The goal is to improve the adaptive unknown inputs observer of [12] so that we recover the signal $m(t)$ with better precision. A first idea would be to compare $\hat{m}(t)$ to 0.5 (1 if $\hat{m}(t) > 0.5$ and 0 else). The problem with such a procedure is that it leads to a lot of errors. It is due to the convergence time at each bit change. Let us say for example that between $6s$ and $7s$ the bit is 0 and between $7s$ and $8s$ the bit is 1. Because of the convergence time, there will be a period at which even if the signal goes to 1, the value of $\hat{m}(t)$ is less than 0.5 (figure 3.5). To avoid this problem, I constructed a modified version of the adaptive unknown inputs observer of [12]. The modified equations are given by 3.23.

$$\begin{cases} \dot{z} = Nz + Jy + Hf(\hat{x}) + Hg(\hat{x})\hat{m}_1 + \frac{1}{2}\hat{\beta}HM(Ty - C_1\hat{x}) \\ \hat{x} = z - Ey \end{cases} \tag{3.23}$$

Figure 3.5: comparison of $\hat{m}$ to 0.5

with the adaptation laws 3.24[2]

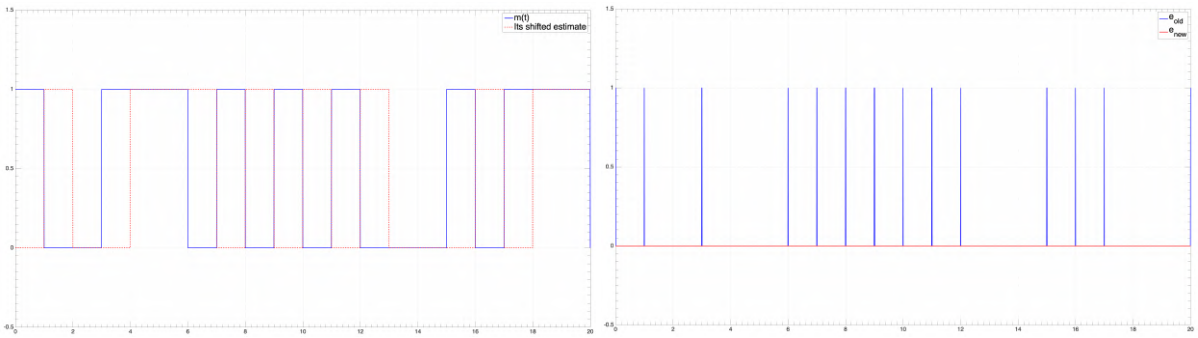
$$\begin{cases} \dot{\hat{m}}_1 = \delta g(\hat{x})^T M(Ty - C_1\hat{x}) \\ \dot{\hat{\beta}} = \gamma |M(Ty - C_1\hat{x})|^2 \\ \hat{m}_2[n] = \dfrac{sign(\frac{1}{\theta_1}\int_{n-\theta_1}^{n} \hat{m}_1(t)dt - \theta_2) + 1}{2}, \quad n = \theta_1, 2\theta_1, \ldots \\ \hat{m}_3(t) = \mathscr{F}(\hat{u}_2, t) \\ \hat{m}(t) = \hat{u}_3(t + \theta_1) \end{cases} \tag{3.24}$$

where $\theta_1, \theta_2 \in \mathscr{R}^+$, and $\mathscr{F}$ is a function defined by 3.25.

$$\mathscr{F}(\hat{m}_2, t) = \begin{cases} 0, & t \in [0, \theta_1) \\ \hat{m}_2[i], & t \in [i, i + \theta_1) \end{cases} \tag{3.25}$$

The signal $\hat{m}_1$ is the unknown input estimate of the original observer. Let us say for example that each second a bit is generated at the input and let us compute $\int_{5-4}^{5} \hat{m}_1(t)dt$. If the result is 1, it means that the bit between $4s$ and $5s$ is 1. If it is 0 then the bit is 0. Because of the convergence time, and the errors in $\hat{m}_1$, the integral can be different

[2]One thing that can happen but is unlikely is that $\hat{m}_2[n] = 1/2$, when the integral divided by θ_1 is exactly equal to θ_2. To prevent this from happening, we can add a condition of the form: If $\hat{m}_2[n] = 1/2$, then $\hat{m}_2[n] = 1$.



(a) The unknown input $m(t)$ and the shifted estimate $\hat{m}_3(t)$ (b) The old error and the new error

Figure 3.6: The simulations of the modified observer

from 1 or 0. For this reason, I compare the result to θ_2. In our example, we can take $\theta_2 = 0.5$. If the integral is greater than 0.5 then it is 1, else it is 0. To achieve this if condition, I use the function $sign$ to which I add 1 and then I divide the result by 2 to obtain a binary sequence. I convert the binary sequence $\hat{m}_2$ into a binary signal using the function $\mathscr{F}$, and the result is the input of the system, but shifted to the right by θ_1. The reason for this delay is that if I want to compute $\hat{m}_2[\theta_1]$ for example, I can not do it at $t = 0$, I need to have the signal $\hat{m}_1$ between 0 and $\theta_1 s$, which causes a delay of $\theta_1 s$. The reason why I divide the integral by θ_1 is that if the integral is done on a period of time different from $1s$, then the result will not coincide with the value of the bit, because the integral gives us the surface, so we need to divide the result by the period of integration to find the amplitude of the signal. The results of the simulations are shown in figure 3.6.

There are two important questions to answer. First of all, are the errors of the original observer really problematic? To answer this question, I masked a plaintext by XORing it with $m(t)$, then I XORed the result with the estimate $\hat{m}(t)$ of the original observer compared to the value 0.5. The result is shown in figure 3.7. As we can see, the plaintext retrieval is noisy because of the errors due to the convergence times. The second question is: can we do without integration? One of the improvements to the original observer that I thought of is to add a sample and hold circuit and compare its output to the value 0.5 as I did for the estimate of the original observer. The reason why I preferred to use the integration is that in the presence of large oscillations in the original estimate, the sample-and-hold-comparator system will lead to errors because if for example at the time of sampling the oscillation goes down below 0.5, even if these oscillations are around 1, the output value of the sample-and-hold-comparator system will be 0. In the integrator system, it is possible to compensate for these oscillations by taking the average of these,

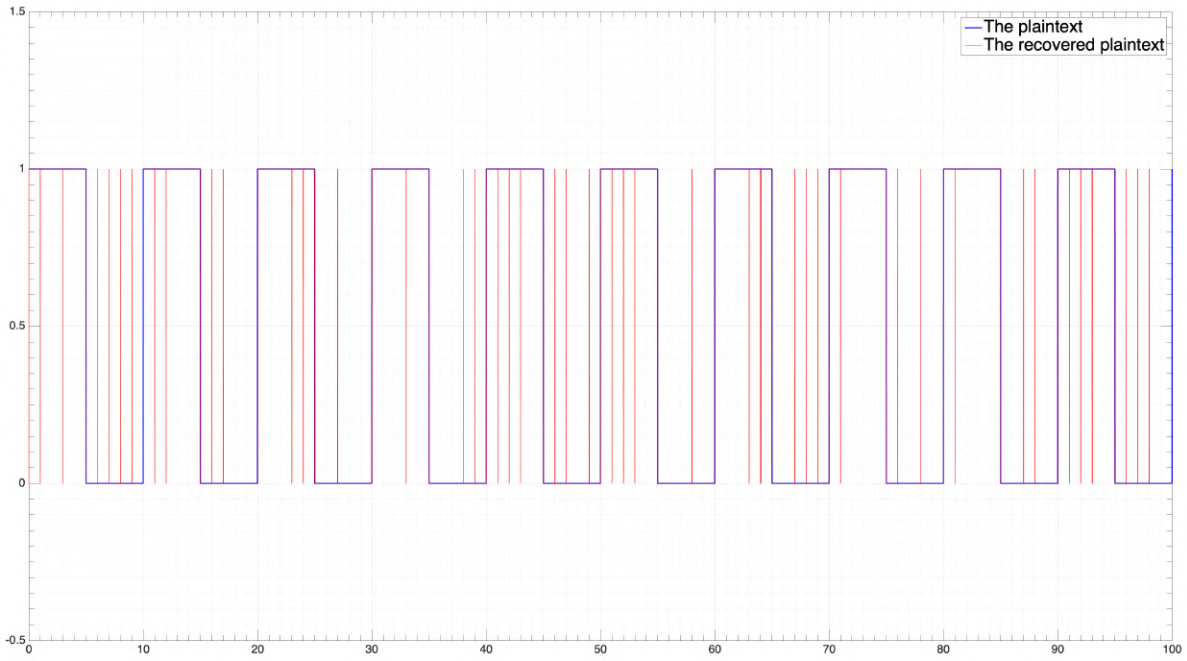


Figure 3.7: The plaintext and the recovered plaintext

which makes it much more robust than the sample-and-hold-comparator system. We can of course try to add a filter before the sample-and-hold-comparator system, but this filter has to be adapted depending on the unknown input frequency and the higher frequency oscillations in the estimate, which is not practical compared to an integrator circuit that works independently of the frequencies of the different signals.

3.2.4 The Tent map based stream cipher

Stream ciphers are cryptosystems used for the rapid encryption and decryption of information of non-predetermined size. The stream cipher I designed is shown in figure 3.8. The Tent map is a chaotic system given by 3.26.

$$x_{n+1} = 1 - r|x_n - 0.5| \tag{3.26}$$

where r is a positive real constant that determines if the system is chaotic or not.

At the emitter level, there are 34 tent maps whose outputs are compared to 0.5. The signal emitted by the first tent map is sent directly to the rest of the emitter. The signal sent by the second tent map is delayed by $1/2s$ and sent to the rest of the emitter. The signal sent by the third tent map is delayed by $1/(2^2)s$ then sent to the rest of the emitter,

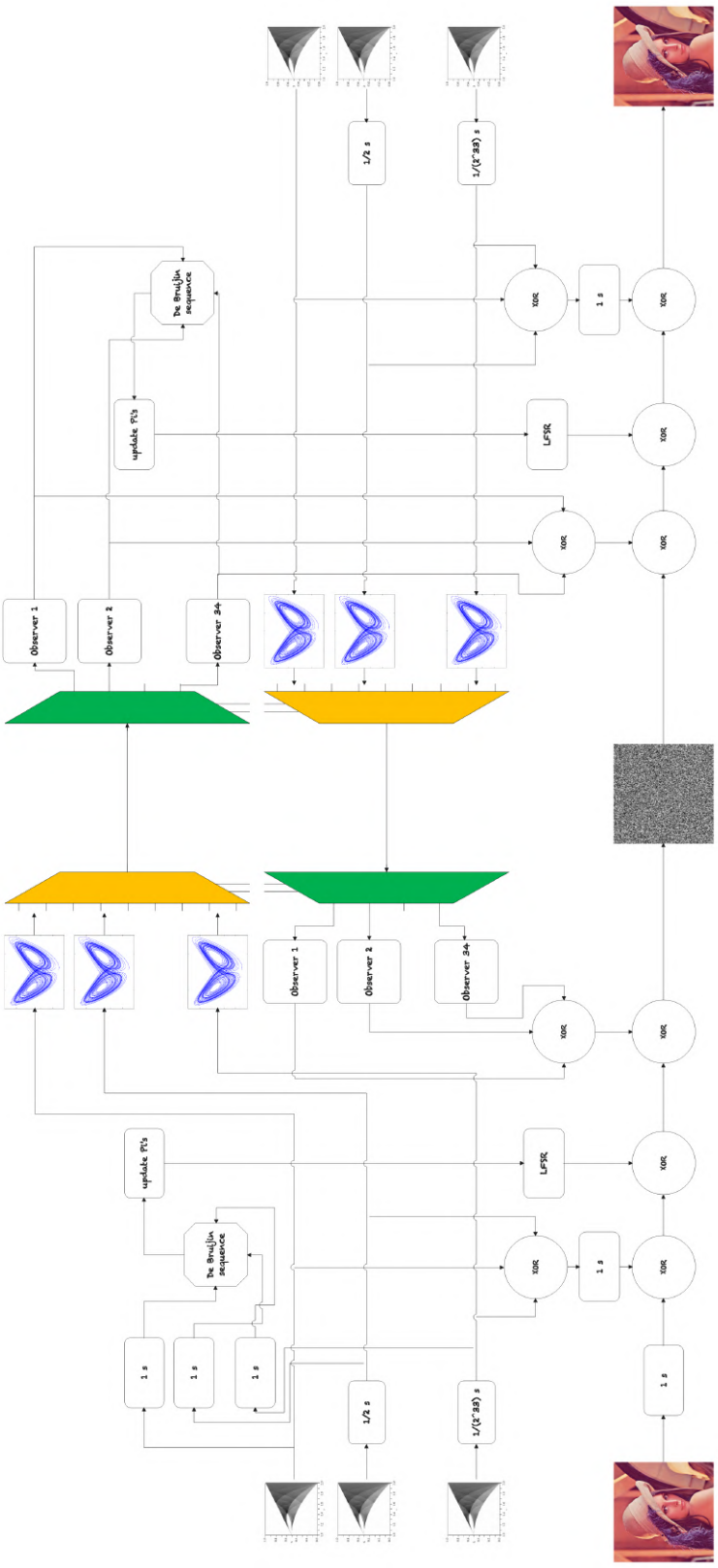


Figure 3.8: The Tent map based stream cipher

and so on until the 34^{th} tent map whose signal is delayed by $1/(2^{33})s$ before being sent to the rest of the emitter. These 34 signals are XORed and the result is delayed by $1s$ and then XORed with the plaintext. When we start using the cryptosystem, we need to wait $1s$ before sending the real plaintext bits for encryption, during this $1s$, we can send 0's through the cryptosystem. The result after XORing the tent maps with the plaintext is a sequence s_i. The 34 tent map signals are each sent as an external input to a chaotic system. These 34 chaotic systems can be different. For example, I chose to use a Rossler system for each of the 34 chaotic systems, the difference between these systems being the parameters of these and the initial conditions. The outputs of these Rossler systems are sent via a multiplexer to the receiver. The second part of the emitter consists of encryption using a variable key LFSR. The key of this LFSR is modified according to an algorithm based on the De Bruijin sequence and on the outputs of the different tent maps of the emitter. The sequence obtained after XORing s_i with the LSFR is denoted by w_i. Finally, the emitter receives from the receiver 34 chaotic signals each corresponding to a chaotic system with an external input located at the receiver. Each one of these signals is then entered into a modified adaptive unknown inputs observer and the 34 reconstructed unknown inputs are XORed with each other then XORed with w_i to obtain the ciphertext r_i.

At the receiver level, we first recover the 34 unknown inputs sent by the emitter through Rossler systems using the modified adaptive unknown inputs observers. These 34 unknown inputs retrieved and shifted by $1s$ (due to the structure of the observer) are XORed between them and then the result is XORed with the ciphertext r_i to obtain a sequence denoted by l_i. They are also used with the De Bruijin sequence to update the LFSR key of the receiver which is the same as that of the emitter. The result of this LFSR is then XORed with l_i to obtain a sequence denoted by v_i. Finally, there are 34 tent maps at the receiver level, each shifted as at the emitter level, and they are all XORed to each other and the result is delayed by $1s$ and XORed with v_i to obtain the recovered plaintext. These 34 signals coming from the tent maps are sent as external inputs to 34 Rossler systems at the receiver, and the result will be sent to the emitter via a multiplexer.

I will now explain in more details three parts of this cryptosystem: the system of 34 tent maps; the algorithm to update the key of the LFSR and finally the key used for this symmetric cipher.

3.2.4.1 The system of 34 tent maps

Because of the convergence time of the observer at each bit change, a very high-frequency bit change in the unknown input of the Rossler system will not be detected, even if we increase the observer gains. This is problematic because we need to encrypt a large number of bits per second, and if we can only generate one bit each second for the encryption then the cryptosystem has no practical use. For this reason, I thought of using multiple tent maps each one delayed by $1/2^n$ so that when we XOR them, the result will look like a bitstream generator that generates a large number of bits each second. This sequence can then be reconstructed at the receiver because each observer will reconstruct the output of one delayed tent map, we just need to XOR them to find the sequence used for encryption at the emitter. Of course, the observer will create a delay of $1s$ in the unknown inputs, it is one of the reasons why we have delays when we XOR the plaintext with the tent maps. It is clear that with 1 tent map, we encrypt 1 bit/s. With 2 tent maps we encrypt 2 bits/s. With 3 tent maps, we encrypt $2^2 = 4$ bits/s, and with 34 tent maps we encrypt 2^{33} bits/s = 1073 MByte/s.

3.2.4.2 How to update the key of the LFSR ?

As we have seen in the state of the art, one way of breaking a LSFR is to use a sufficient number of pairs of (input, output) so that we can recover the $p_i's$ by solving a linear system. My idea here is to continuously update the $p_i's$ so that Oscar can not compute them by solving linear systems. The way I update the $p_i's$ is by using an algorithm based on the De Bruijin sequence.

Let us denote each one of the 34 tent maps by the symbol T_i where i is the number of the tent map. We thus created an alphabet composed of 34 symbols 3.27.

$$A = \{T_1, T_2, ..., T_{34}\} \tag{3.27}$$

A De Bruijin sequence of order n on A is a cyclic sequence in which every possible length-n string on A occurs exactly once as a sub-string. The total number of distinct De Bruijin sequences of order n on an alphabet of k symbols is given by 3.28.

$$B(k,n) = \frac{(k!)^{k^{n-1}}}{k^n} \tag{3.28}$$

For example, on the alphabet $\{0,1\}$ there are exactly 2 De Bruijin sequences of order 3 : 00010111 and 11101000.

My goal is to update the $p_i's$ each $1/2^{33}s$. To achieve this, I will use De Bruijin sequences of order 2^{33} on A. The number of distinct De Bruijin sequences of order 2^{33} on A is given by 3.29.

$$B(34,2^{33}) = \frac{(34!)^{34^{2^{33}-1}}}{34^{2^{33}}} \tag{3.29}$$

The LSFR is designed such that the number of $p_i's$, let us denote it by m, is less than $B(34,2^{33})$. We select then m distinct De Bruijin sequences from the existing $B(34,2^{33})$. For each p_i we have a sequence of 2^33 bits updated each second. The value of each p_i is updated each $1/2^{33}$ second using the corresponding De Bruijin sequence, as it is shown in figure 3.9. For example, if the sequence corresponding to p_1 is $\{T_1,T_4,...,T_3,T_2\}$, then p_1 is equal to T_2 for the first $1/2^{33}s$, then it is equal to T_4 for the next $1/2^{33}s$ and so on. This process of updating the $p_i's$ begins after $1s$ of starting the tent maps. The value of the $p_i's$ before that is not important because the real encryption begins also $1s$ after starting the tent maps.

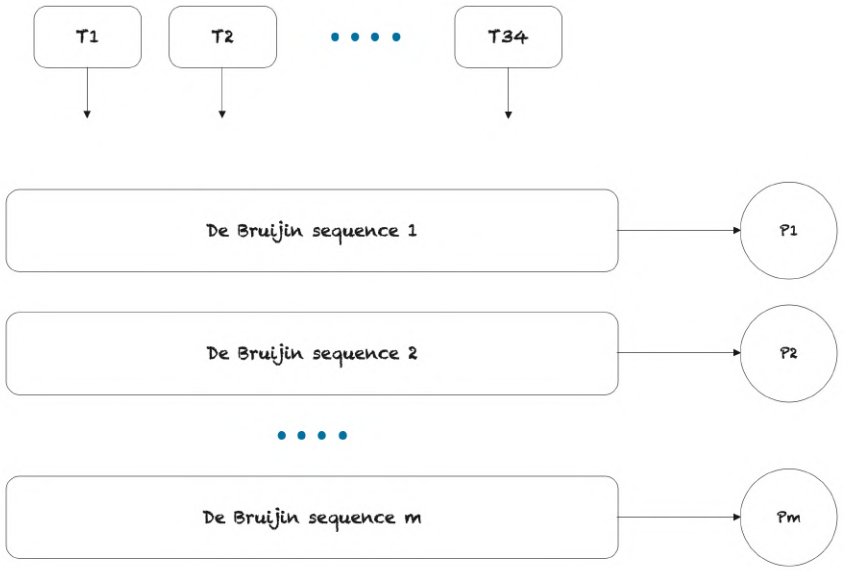


Figure 3.9: The way to update the key of the LFSR

3.2.4.3 The key of the tent map based stream cipher

The tent map stream cipher is a symmetric cryptosystem. The key[3] is given by 3.30.

$$key = (a_1;...;a_{34};b_1;...;b_{34};c_1;...;c_{34};r_1;...;r_{34};R_{1,1};...;R_{34,3};TM_1;...;TM_{34}) \quad (3.30)$$

where a_i, b_i, c_i are the parameters of the i^{th} Rossler system, r_i is the parameter of the i^{th} tent map, $R_{i,j}$ is the j^{th} initial condition of the i^{th} Rossler system and TM_i is the initial condition of the i^{th} tent map. As we can see, they have the same initial conditions at the emitter and at the Receiver. If the chaotic systems were programs in C for example, this would have had no sense because the emitter and the receiver would cancel themselves and the cryptosystem would become a simple LFSR, but because the chaotic systems are chaotic electronic circuits, we can choose the same parameters, because we are sure that noise will affect the initial conditions and modify completely the behavior of the chaotic systems. Of course, the beginning of the chaotic systems of the emitter and the ones of the receiver will be the same for a short period of time. During this period, the encryption will be done using the LFSR with a variable key. After this period of time, the chaotic signals will become different and they will affect directly the plaintext with the XOR gates[4].

3.2.5 The security of the tent map stream cipher

I have generally described the tent map-based stream cipher. Without more details, this cryptosystem cannot be really analyzed, and cannot be considered for real practical use. This is why in future work it will be interesting to present the tent map stream cipher in more detail. However, I found it interesting to present a security flaw of this stream cipher which is also present in several chaotic cryptosystems presented in the literature [1, 38]. There are a lot of attacks based on the low-frequency characteristics of certain chaotic systems, including the Rossler system in particular. In general, using chaotic masking or chaotic modulation (like in my stream cipher) can be dangerous if done with bad chaotic systems like the Lorenz system or the Rossler system. The reason for this is that it is usually enough to use a simple filter to find the unknown input because the

[3] I did not add the matrices of the observers as part of the key. The reason for this is that in a lot of observers, changes in certain values of the matrices will not affect completely the convergence, which can lead to security flaws if the matrices are part of the key. Instead, there is at the emitter and the receiver algorithms that compute the matrices gains given the chaotic system's parameters.

[4] We can also wait until the chaotic signals become different before we start the encryption. Another possibility is to assume that we use use different values for the chaotic systems at the emitter and the chaotic systems at the receiver.

frequency of this one is easily distinguished from the frequency of the Rossler, Lorenz, and the other "bad" chaotic systems. There are two things we can do to remedy this problem. Either we use chaotic systems with good frequency characteristics, which risks greatly reducing the number of chaotic systems we can use. Either we find a way to make systems like Rossler and Lorenz more "frequency acceptable". I took the second approach. So, in this part I will show through simulations why the classic Rossler system is a bad choice of chaotic system for cryptography, then I will present a modification that will improve the frequency characteristics of the Rossler system. Finally, we will see that this modification can easily be generalized to other chaotic systems such as the Lorenz system.

The main security problem of the Rossler-based encryption schemes is that it is possible to recover the unknown input using the Rossler system outputs. To achieve this, we can use a filter for example. This approach is used in many cryptanalysis articles [1, 38]. Here, I will train an ANFIS observer[5] to reconstruct the external input of the Rossler system using the signals x_1, x_3 and $\dot{x}_3$. To obtain $\dot{x}_3$, we can use a differentiator circuit. The reason why I use $\dot{x}_3$ is that the external input of the Rossler system affects it directly, in the sense that we can see some features of the external input inside the signal $\dot{x}_3$ as shown in figure 3.10.

To train the ANFIS observer, I used as an output to the ANFIS a pseudo-random binary sequence. This sequence is generated using the "random numbers" SIMULINK block whose output is entered in the "compare to 0" SIMULINK block. I used as an input the signals x_1, x_3 and $\dot{x}_3$. I used the SIMULINK block "to workspace" with a sampling time of $0.005s$ to collect the data during $20s$ of simulation. I trained the ANFIS using the "anfisedit" command in MATLAB. I used the default settings of "anfisedit" and I used for all the signals Gaussian membership functions "gaussmf" and I did 20 epochs. The results of the simulations are shown in figure 3.11.

The reason why this attack worked is because of the bad frequency properties of the Rossler system, especially of the signal $\dot{x}_3$. One way of overcoming this problem is to use a chaotic system with a spectrum that is infinitely broad, flat, and of much higher

[5]Both cryptanalysis techniques have their advantages and disadvantages. With ANFIS for example, Oscar will need to have the chaotic system at his disposal for a certain time. For filters, it will be necessary to compute the frequency characteristics of the chaotic system and see if there is a frequency that stands out from the others. The reason why I chose ANFIS is just to introduce this cryptanalysis technique as a possible replacement for classic filtering techniques. More in-depth work must be done to find out whether cryptanalysis by ANFIS really offers advantages over filtering techniques.

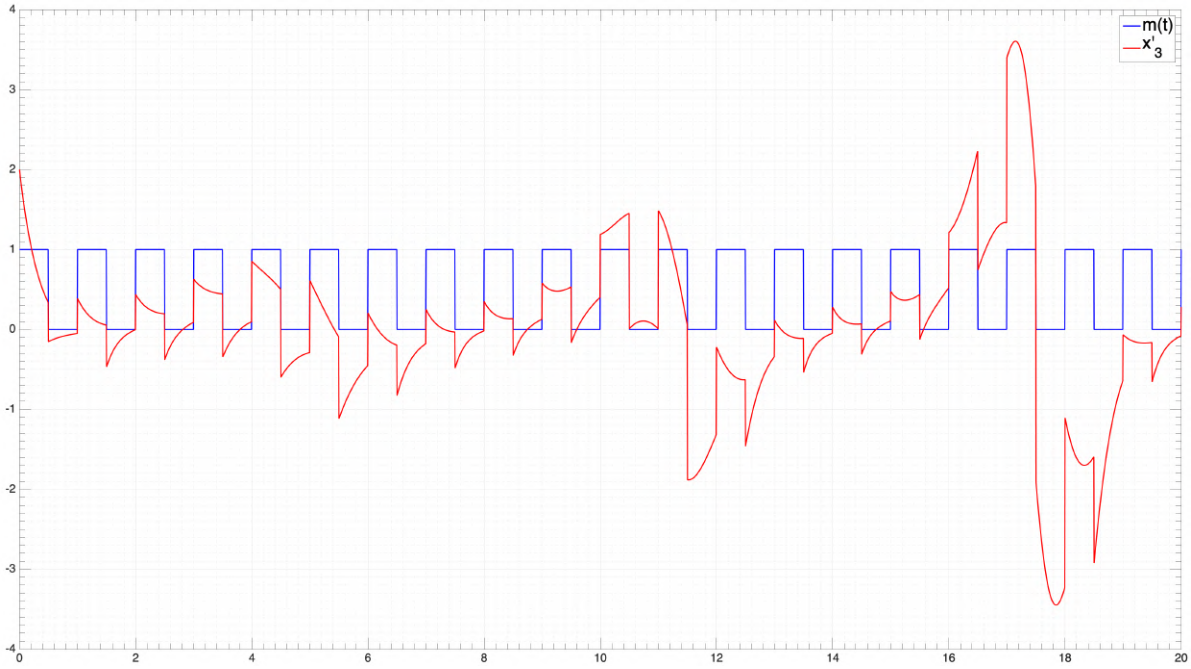


Figure 3.10: The signal $\dot{x}_3$ and the unknown input $m(t)$

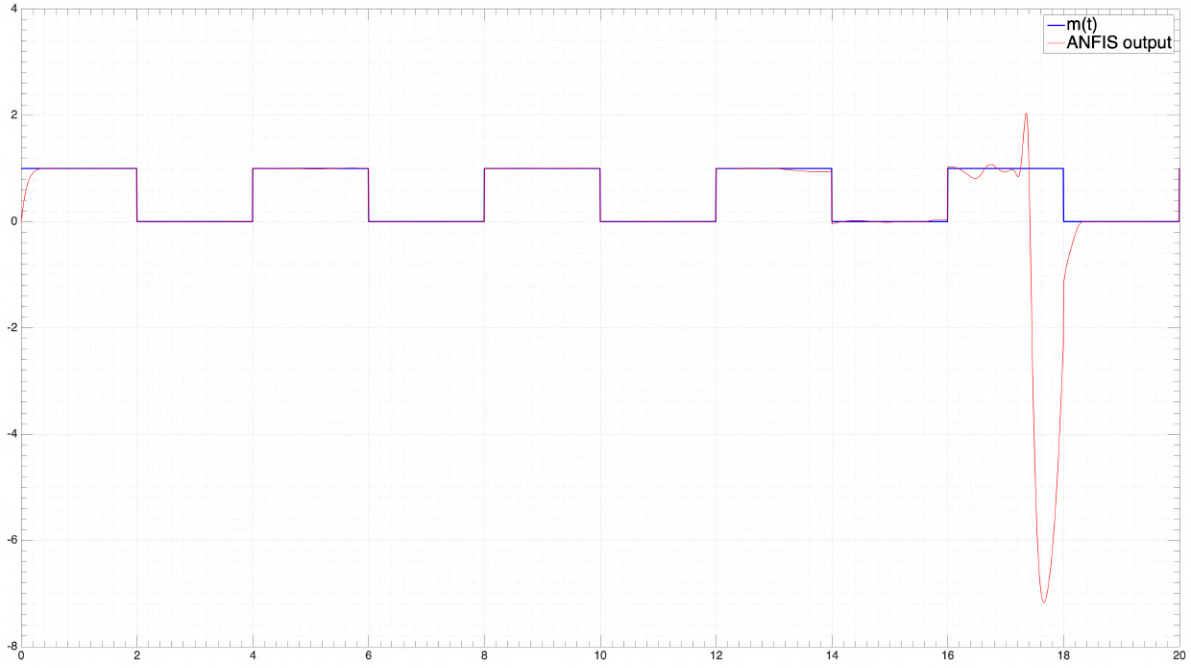


Figure 3.11: Cryptanalysis of the classical Rossler system using ANFIS

power density than the signal to be concealed [2]. I designed a modification of the Rossler system which has better frequency properties than the classical Rossler system. This modified system is given by 3.31.

$$\begin{cases} \dot{x}_1 = -(x_2 + x_3) + d(t) \\ \dot{x}_2 = x_1 + a x_2 + d(t) \\ \dot{x}_3 = b + x_3(x_1 - c) + u(t) x_3 s_2(t) + k s_1(t) + d(t) \\ y_1 = x_1 + 2d(t) \\ y_2 = x_3 + d(t) \end{cases} \tag{3.31}$$

where $s_1(t)$ and $s_2(t)$ are pseudo-random binary sequences, and $k \in \mathbf{R}$. This system can be rewritten in matrix form 3.32.

$$\begin{cases} \dot{x} = \begin{bmatrix} 0 & -1 & -1 \\ 1 & a & 0 \\ 0 & 0 & -c \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} (b + x_1 x_3 + k s_1(t)) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} x_3 s_2(t) m(t) + \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} d(t) \\ y = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} x + \begin{bmatrix} 2 \\ 1 \end{bmatrix} d(t) \end{cases} \tag{3.32}$$

As we can see, the only difference with 3.20 is the functions f and g which are now equal respectively to $(b + x_1 x_3 + k s_1(t))$ and $x_3 s_2(t)$. For this reason, the matrices of the adaptive unknown inputs observer are the same as before.

I set $k = 10$, and $s_1(t) = s_2(t)$ a pseudo binary sequence constructed in the same way that the one used for the training of ANFIS. The unknown input $m(t)$ is a binary signal starting with 1 and shifting the bit each $0.5s$. The attractor of the modified Rossler system is shown in figure 3.12.

As we can see in figure 3.13, the frequency properties of $\dot{x}_3$ are better. We can improve them more by changing k, $s_1(t)$ and $s_2(t)$.

We can try the ANFIS-based attack. I trained an ANFIS using the same pseudo random binary sequence as before, the same number of data points, the same settings and the same number of epochs. The result of the simulation is shown in figure 3.14. As we can see, the Rossler system hides now the unknown input better. My idea is to add a controlled noise to the derivative of the state vector, so that we make the variation of the state vector more random. We can test this technique of adding a controlled noise to

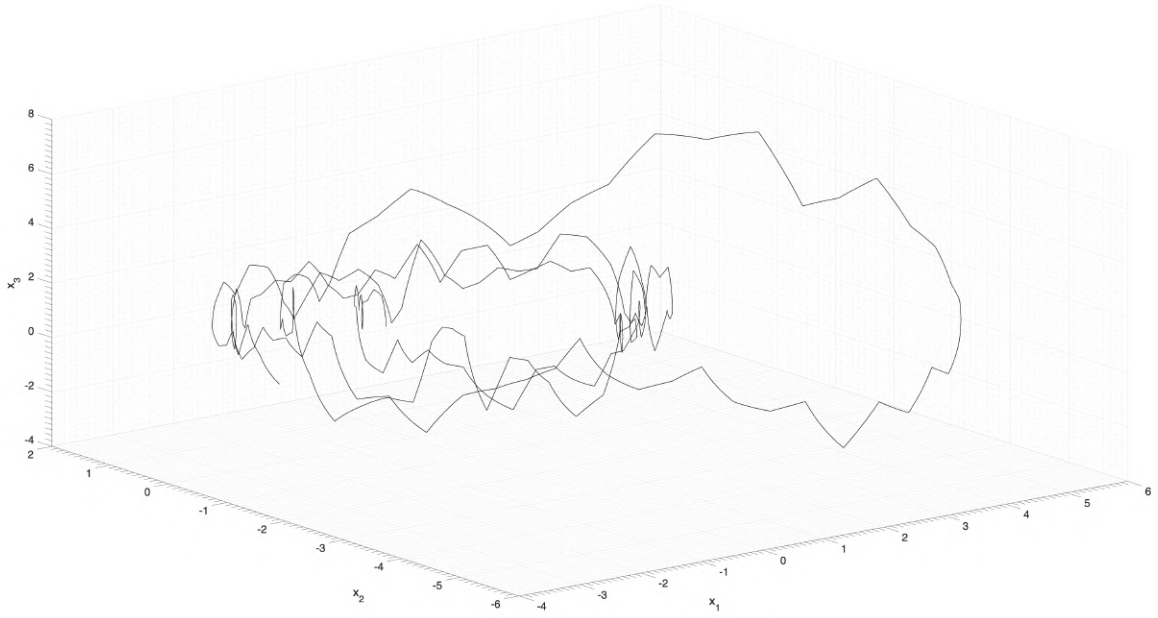
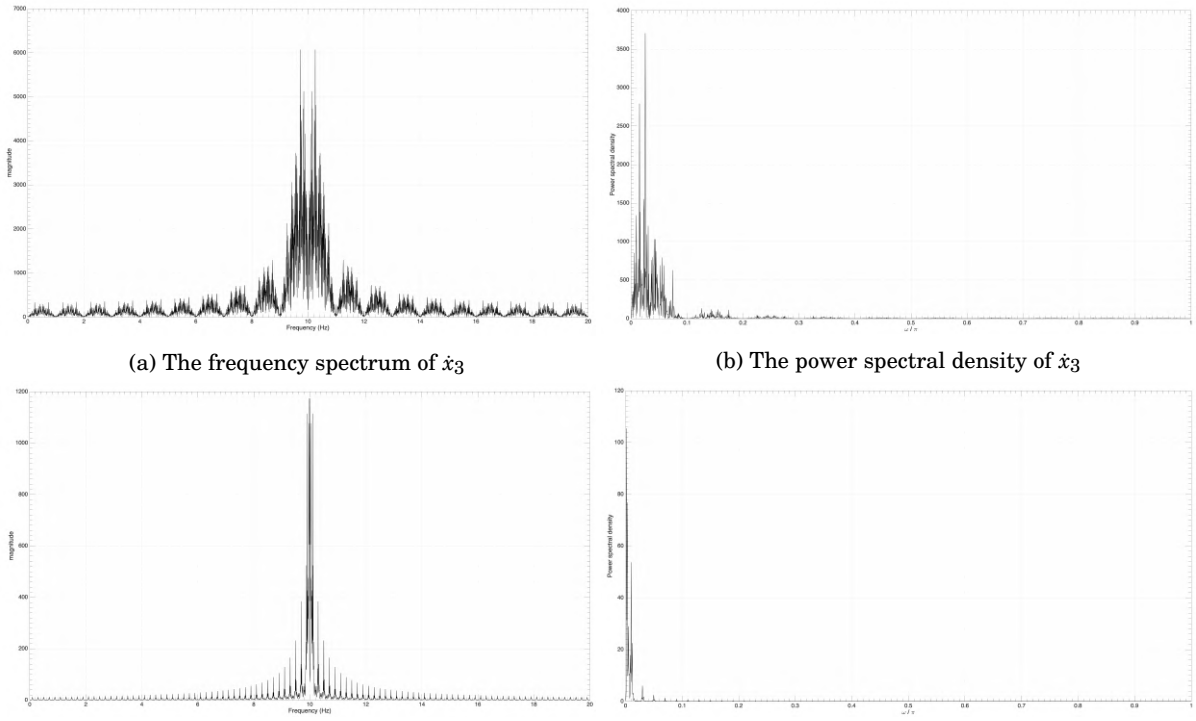


Figure 3.12: The attractor of the modified Rossler system



(a) The frequency spectrum of $\dot{x}_3$

(b) The power spectral density of $\dot{x}_3$

(c) The frequency spectrum of the original $\dot{x}_3$

(d) The power spectral density of the original $\dot{x}_3$

Figure 3.13: The frequency characteristics of $\dot{x}_3$

Figure 3.14: Cryptanalysis of the modified Rossler system using ANFIS

the derivatives of the states on a Lorenz system. The modified Lorenz attractors that I obtained are shown in figure 3.15.

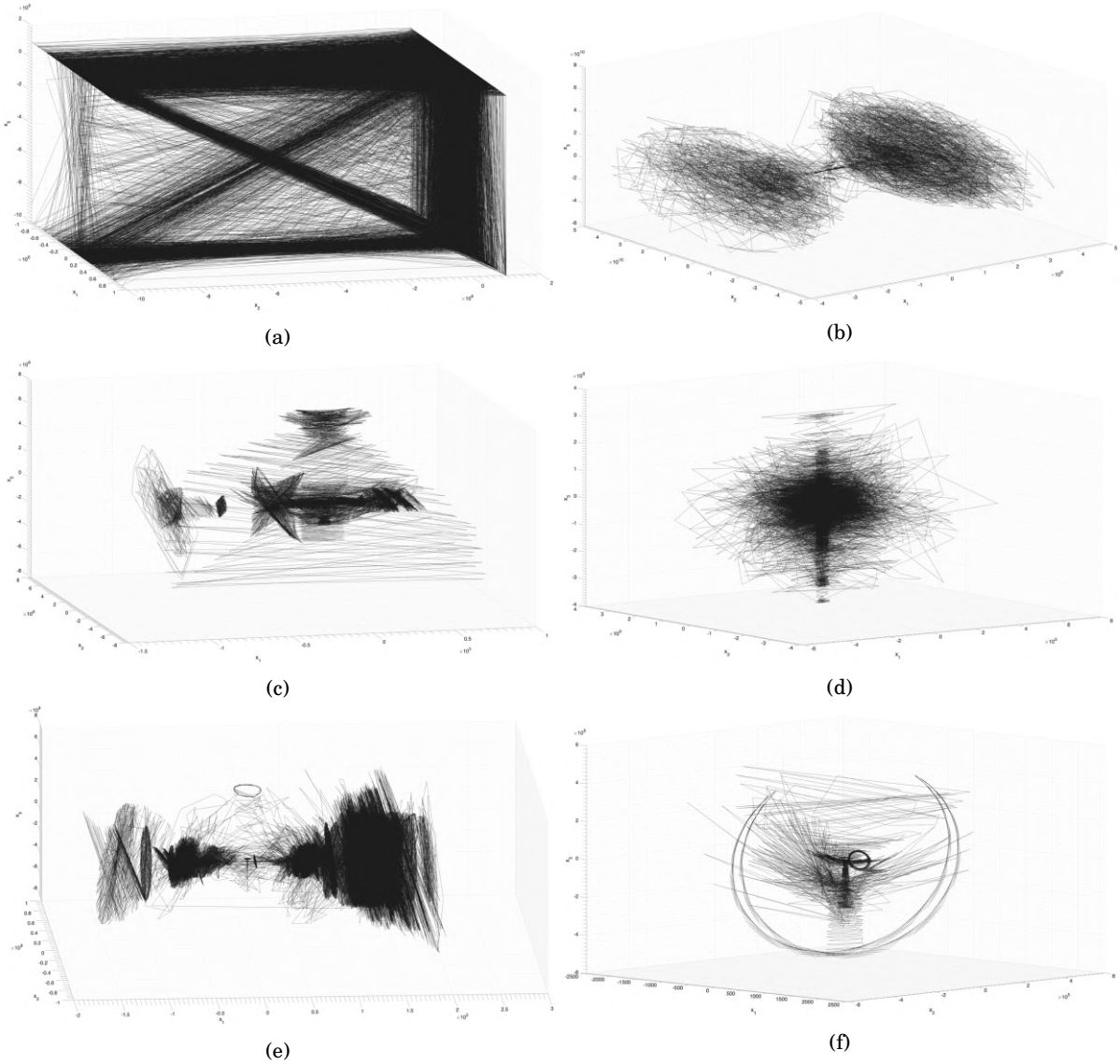
I obtained these attractors using variants of 3.33.

$$\begin{cases} \dot{x}_1 = 5\sigma(x_2 - x_1 s)s \\ \dot{x}_2 = x_1(\rho - x_3) - x_2 s \\ \dot{x}_3 = x_1 x_2 - 10\beta x_3 s \end{cases} \tag{3.33}$$

where $\rho = 28$, $\sigma = 10$, $\beta = 8/3$ and s is a random sequence of numbers generated using the "random numbers" SIMULINK block. For example, we can see in figures 3.16 and 3.17 the sensitivity to initial conditions of the attractor shown in figure 3.15b.

3.3 Adaptive sliding mode unknown inputs observer

I present in this section an adaptive sliding mode unknown inputs observer which was built-in [12]. The main advantage of this observer compared to the previous one is that it allows us to predict much better the unknown inputs with derivatives almost always zero and which vary at high frequency, as well as the signals whose derivative is not necessarily almost always zero. I will use the observer in this section mainly to estimate



(a) (b) (c) (d) (e) (f)

Figure 3.15: The improved Lorenz attractors

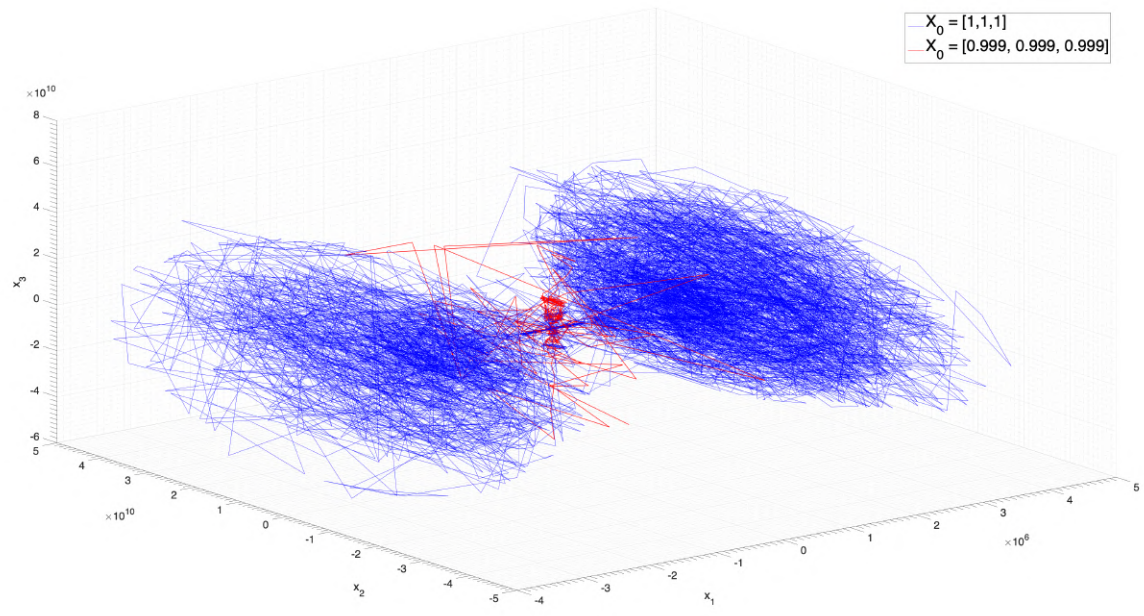


Figure 3.16: The sensitivity to initial conditions of the modified attractor

the first type of signal. We will see that the observer presents a certain number of errors in the estimation, these being mainly due to the convergence time at each value change, as for the first observer and to the noise in the output of the system. I would therefore present a technique inspired by the one used in the first observer to cancel these errors. I will end this part by introducing a chaotic cryptosystem for the transmission of audio signals.

3.3.1 The original observer

The adaptive sliding mode unknown inputs observer constructed in [12] is for systems of the form 3.34.

$$\begin{cases} \dot{x}_\star = A_0 x_\star + B f_0(x_\star) + F\eta_1(t) \\ y = C_0 x_\star + G_0 \eta_2(t) \end{cases} \tag{3.34}$$

where $x_\star \in \mathbf{R}^n$ is the state vector, $\eta_1 \in \mathbf{R}^{q_1}$ is the vector of unknown inputs, $y \in \mathbf{R}^p$ is the output, $\eta_2(t) \in \mathbf{R}^{q_2}$ is an additive noise and f_0 is a $\mathscr{C}^1$ function.

The system has to satisfy the following conditions:
(a) $q_1 + q_2 \leq p$.
(b) F and G_0 are full rank and $rank(C_0F) = rank(F)$.
(c) $\eta_1(t)$ and $\eta_2(t)$ are bounded and their first derivatives are bounded.

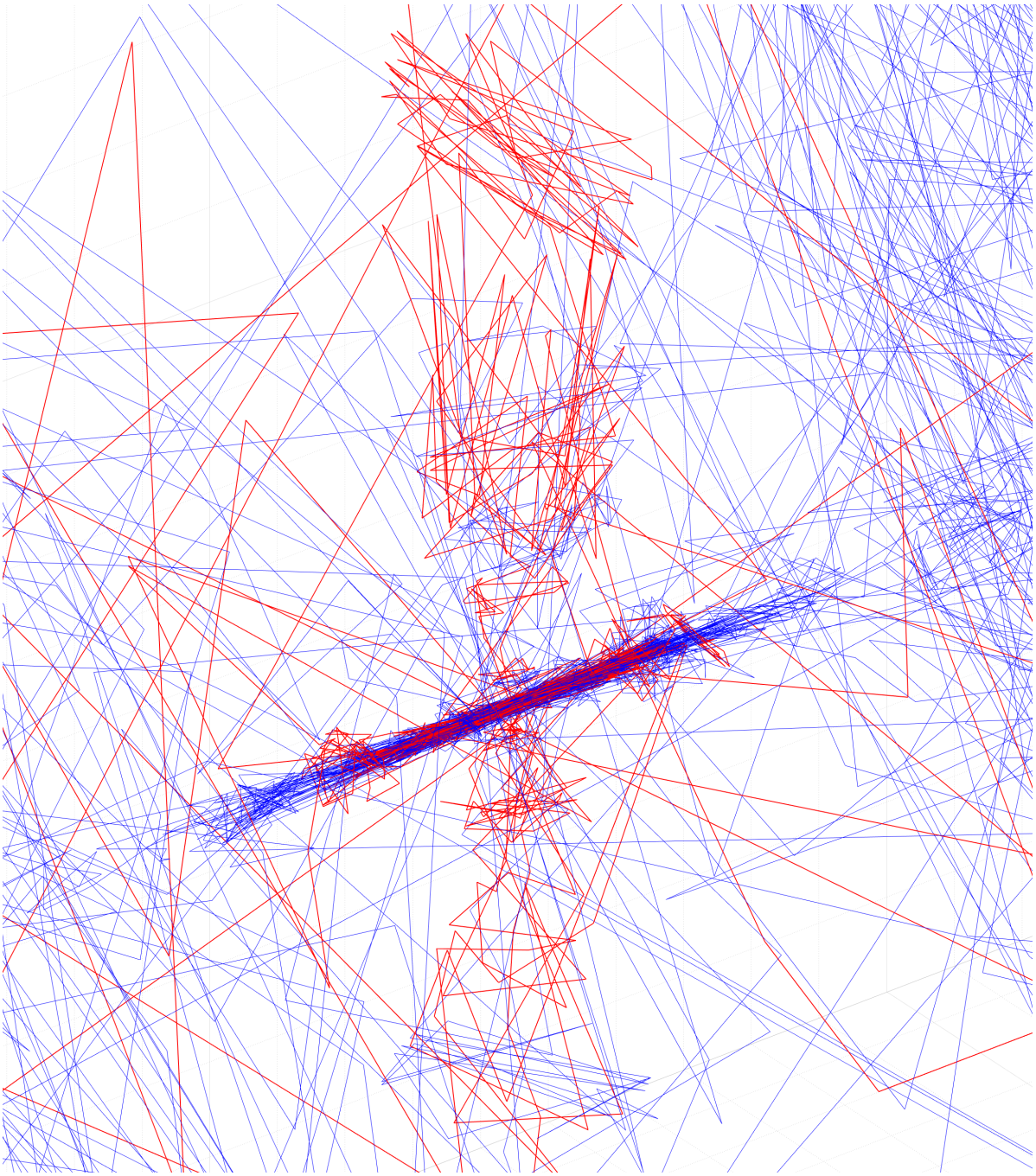


Figure 3.17: Zoom in 3.16

(d) The solutions $x_\star(t)$ are globally uniformly bounded.

We define now the augmented state $x = [x_\star^T, \eta_2^T]^T \in \mathbf{R}^{n+q_2}$, and the matrices :

$$A = [A_0\ 0],\quad C = [C_0\ G_0],\quad T = [I_n\ 0]$$

and the function f that satisfies : $f(x) = f_0(x_\star)$.

Under these assumptions, we can rewrite 3.34 under the singular form 3.35.

$$\begin{cases} T\dot{x} = Ax + Bf(x) + F\eta_1(t) \\ y = Cx \end{cases} \tag{3.35}$$

The adaptive sliding mode unknown inputs observer for 3.35 is given by 3.36.

$$\begin{cases} \dot{z} = Kz + Jy + H_1 f(\hat{x}) + \dfrac{1}{2}\hat{\beta}H_1 M(y - C\hat{x}) + H_2 u \\ \hat{x} = z - Ey \end{cases} \tag{3.36}$$

with the adaptation laws 3.37.

$$\begin{cases} \dot{\hat{\beta}} = \gamma_1 |M(y - C\hat{x})|^2 - \sigma_2\hat{\beta} \\ \dot{\hat{\rho}} = \gamma_2 |S| - \sigma\hat{\rho} \end{cases} \tag{3.37}$$

and with u and S given by 3.38.

$$\begin{cases} u = (GH_2)^{-1}[\delta S + \hat{\rho}\dfrac{S}{\epsilon + |S|} - GPA\hat{x} - GH_1 f(\hat{x}) - \dfrac{1}{2}GH_1\hat{\beta}M(y - C\hat{x})],\quad \epsilon > 0 \\ S = NCe(t) + \displaystyle\int_0^t GLCe(\tau)d\tau = N(y(t) - C\hat{x}(t)) + \int_0^t GL(y(\tau) - C\hat{x}(\tau))d\tau \end{cases} \tag{3.38}$$

where $\gamma_1, \gamma_2, \sigma, \sigma_2, \delta, \epsilon \in \mathbf{R}^+$, $e = x - \hat{x}$ and the matrices $K, J, H_1, H_2, M, E, P, G, L$ and N are obtained by carrying out the following steps:

Step 1: compute P and E using 3.39.

$$X = R_2 R_1^+ - Z_a(I - R_1 R_1^+) \tag{3.39}$$

where Z_a is any matrix of the appropriate dimension (for example, $Z_a = 0$), $X = [P\ E]$, $R_1 = [T^T\ -C^T]^T$ and $R_2 = I_{n+q_2}$.

Step 2 : compute H_1 and H_2 using 3.40 and 3.41.

$$H_1 = PB \tag{3.40}$$

$$H_2 = PF \tag{3.41}$$

Step 3 : choose N and G such that $GH_2 = NCH_2$ is invertible.

Step 4 : compute the matrices A_G and B_G using 3.42 and 3.43.

$$A_G = [I - H_2(GH_2)^{-1}G]PA \tag{3.42}$$

$$B_G = [I - H_2(GH_2)^{-1}G]H_1 \tag{3.43}$$

Step 5 : solve the convex optimization problem 3.44, 3.44, 3.45, 3.46 and 3.47.

$$Min \quad \rho_\star,$$

$$P_G > 0 \tag{3.44}$$

$$P_G A_G + A_G^T P_G + RC + C^T R^T < 0 \tag{3.45}$$

$$\begin{bmatrix} \rho_\star I & B_G^T P_G - MC \\ P_G B_G - C^T M^T & \rho_\star I \end{bmatrix} \geq 0 \tag{3.46}$$

$$L = -P_G^{-1} R \tag{3.47}$$

Step 6 : compute K and J using 3.48 and 3.49.

$$K = PA - LC, \text{ is Hurwitz} \tag{3.48}$$

$$J = L - KE \tag{3.49}$$

The idea behind this observer is that the function u will converge to the unknown input $\eta_1(t)$. It is stated in the following theorem: [12]

For all $\epsilon_\eta > 0$, there exists $\delta, \gamma_1, \gamma_2, \sigma_2$ and $0 < T_\eta < \infty$ such that 3.50 holds.

$$|u(t) - \eta_1(t)| \leq \epsilon_\eta, \quad \forall t \geq t_0 + T_\eta \tag{3.50}$$

3.3.2 Example : The Dimassi's auxilliary dynamical system

In [12], the author used a dynamical system in his cryptosystem called the auxiliary dynamical system. It is given by 3.51.

$$\begin{cases} \dot{x}_t = -x_t + F_t E(t) \\ y_t = x_t + G_0 d(t) \end{cases} \tag{3.51}$$

which is a special case of 3.34 where $x = x_t = [x_{t1}, x_{t2}, x_{t3}]^T \in \mathbf{R}^3$, $y = y_t = [y_{t1}, y_{t2}, y_{t3}]^T \in \mathbf{R}^3$, $\eta_1(t) = E(t) \in \mathbf{R}^2$, $\eta_2(t) = d(t) \in \mathbf{R}$, $A = -I_3$, $B = 0$, $C_0 = I_3$, $f(x) = 0$ and

$$F = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad G_0 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

The observer matrices for this system are given by :

$$E = \begin{bmatrix} -0.3333 & 0.1667 & 0.1667 \\ 0.1667 & -0.3333 & 0.1667 \\ 0.1667 & 0.1667 & -0.3333 \\ -0.3333 & -0.3333 & -0.3333 \end{bmatrix}, \quad H_1 = 0, \quad H_2 = \begin{bmatrix} 0.1667 & 0.1667 \\ 0.6667 & 0.1667 \\ 0.1667 & 0.6667 \\ -0.3333 & -0.3333 \end{bmatrix}, \quad G = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix},$$

$$K = \begin{bmatrix} -0.4172 & -0.2796 & -0.2796 & 0.0237 \\ 0.6924 & -1.2927 & -0.2365 & 0.1632 \\ 0.6924 & -0.2365 & -1.2927 & 0.1632 \\ -0.4048 & 0.3502 & 0.3502 & -0.7043 \end{bmatrix}, \quad J = \begin{bmatrix} -0.2874 & 0.1438 & 0.1438 \\ -0.3190 & 0.1736 & 0.1455 \\ -0.3190 & 0.1455 & 0.1736 \\ 0.2517 & -0.1258 & -0.1258 \end{bmatrix}, \quad M = 0$$

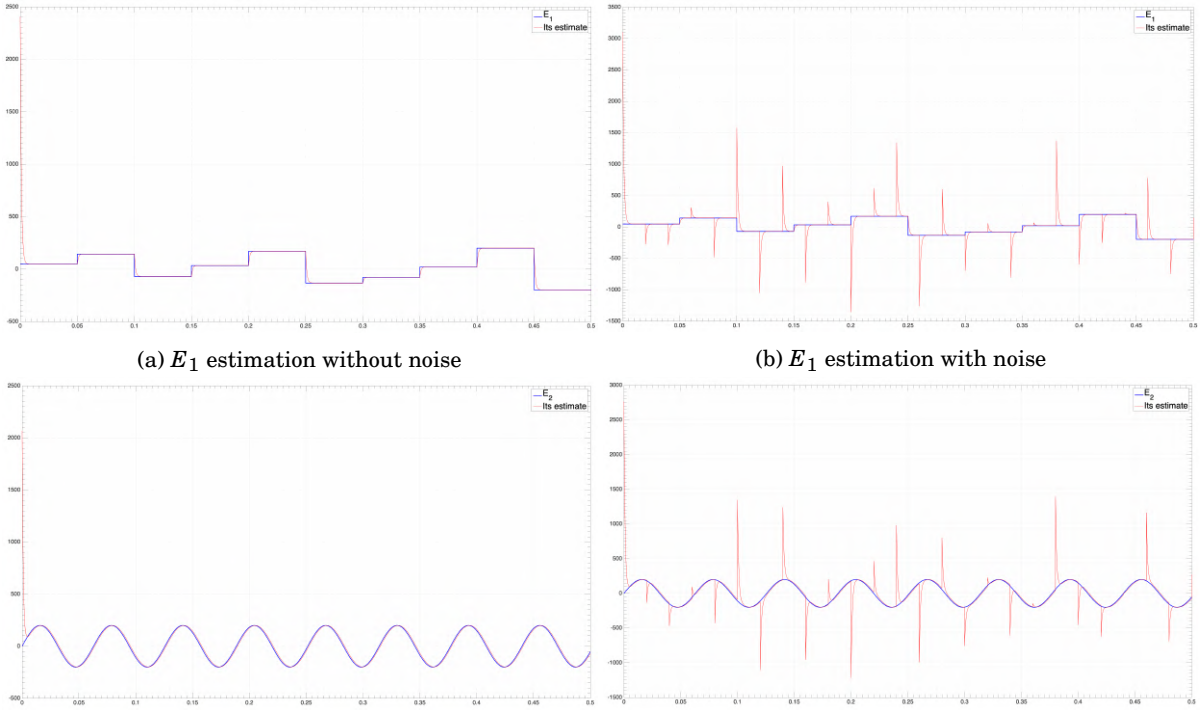
$$P = \begin{bmatrix} 0.6667 & 0.1667 & 0.1667 \\ 0.1667 & 0.6667 & 0.1667 \\ 0.1667 & 0.1667 & 0.6667 \\ -0.3333 & -0.3333 & -0.3333 \end{bmatrix}, \quad N = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad L = \begin{bmatrix} -0.2494 & 0.1130 & 0.1130 \\ -0.8591 & 0.6261 & 0.0699 \\ -0.8591 & 0.0699 & 0.6261 \\ 0.7381 & -0.0169 & -0.0169 \end{bmatrix}$$

I did simulations on MATLAB/SIMULINK for this observer, for given inputs and adaptation gains. In all the simulations I fix $\delta = 1000$, $\epsilon = 0.0001$, $\gamma_2 = 10000$ and $\sigma = 1000$. The initial conditions are set to $x_0 = [-0.11, -0.14, 0.2]$, $\hat{z}_0 = [-0.11, -0.14, 0.2, -0.38]$ and $\hat{\rho}_0 = 0$. The results of the simulations are shown in figure 3.18. The noise[6] I added is shown in figure 3.19.

3.3.3 The modified adaptive sliding mode unknown inputs observer

I will try now to improve the adaptive sliding mode unknown inputs observer of [12] for the estimation of signals whose derivative is almost always zero (like the signal E_1 in the previous part). There are two problems that I will try to solve. The first one is due to the convergence times at each value change in the input. As for the adaptive unknown inputs observer presented in the first section, these convergence times cause errors. The

[6] A smaller noise will have a similar effect on a smaller input.



(a) E_1 estimation without noise

(b) E_1 estimation with noise

(c) E_2 estimation without noise

(d) E_2 estimation with noise

Figure 3.18: The simulations of the original observer

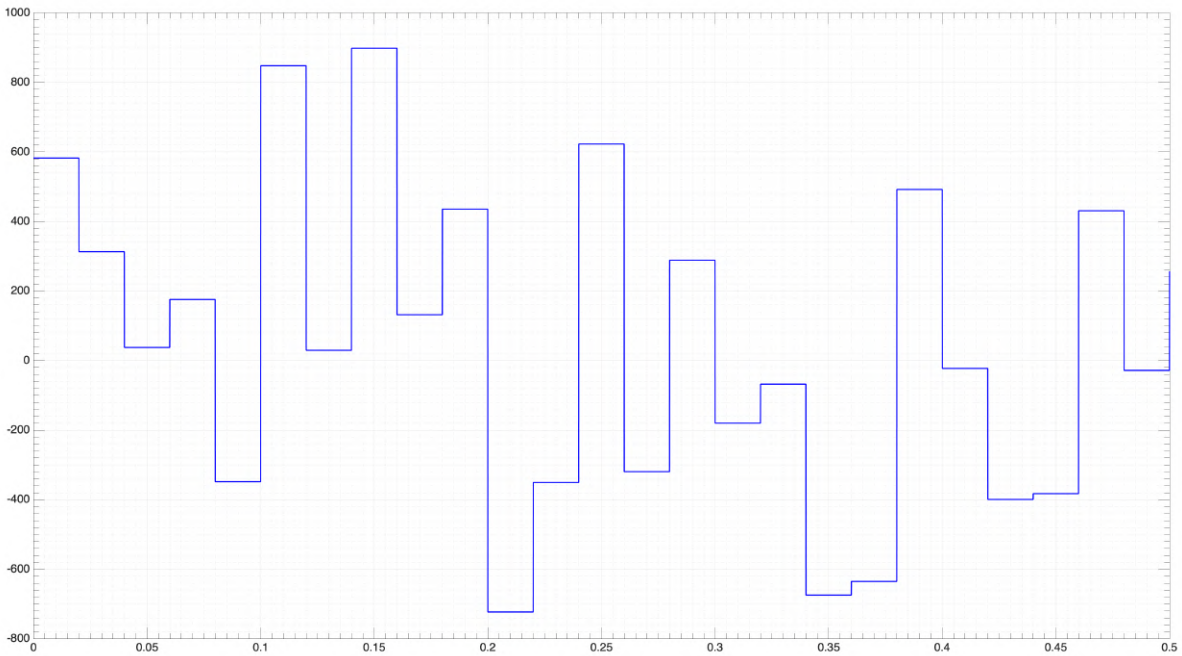


Figure 3.19: The noise $\eta_2(t)$

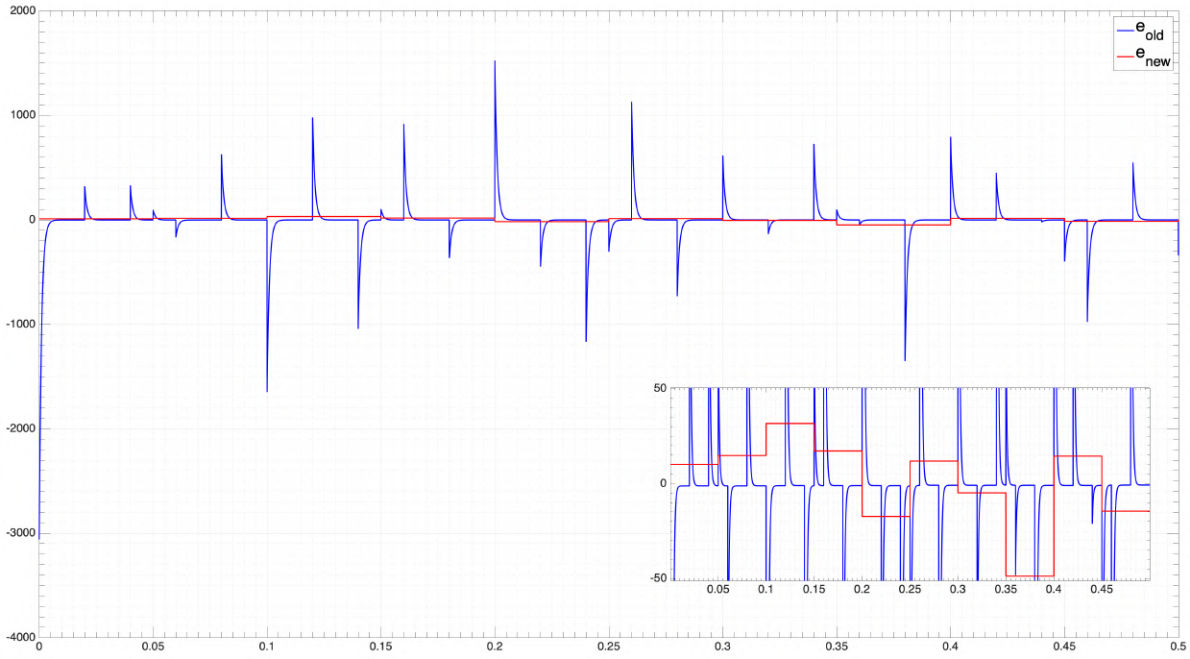


Figure 3.20: The error in the estimation of E_1

second problem is due to the noise η_2 which causes many pics of values in the estimate. To avoid these problems, I propose to add the equations[7] 3.52.

$$\begin{cases} \hat{u}_{11}[n] = \dfrac{1}{\theta_1 - 2\theta_2} \displaystyle\int_{n-\theta_1+\theta_2}^{n-\theta_2} u(\tau)_1 d\tau, \quad n = \theta_1, 2\theta_1, \ldots \\ \hat{u}_{21}(t) = \mathscr{F}(\hat{u}_{21}, t) \\ \hat{E}_1(t) = \hat{u}_{21}(t + \theta_1) \end{cases} \tag{3.52}$$

where $\theta_1, \theta_2 \in \mathbf{R}^+$. To explain this modified observer, let us assume that E_1 has a new value each $0.05s$, and let us take $\theta_1 = 0.05s$. Let us also assume that $\eta_2 = 0$, so that u_1 is like in figure 3.18a, i.e without high pics between each $0.05s$. I want to estimate the value of E_1 between 0 and $0.05s$. I chose θ_2 greater than the convergence time (in our example, I can take $\theta_2 = 0.01s$). I integrate the signal u_1 from $0.01s$ to $0.04s$, and then I divide the result by $\theta_1 - 2\theta_2$ to find the amplitude of the signal. The result of the simulation is shown in figure 3.20.

[7] As for the first observer of this chapter, this technique can be used for other observers. It is a second layer of estimation that decreases the error before using the recovered signal

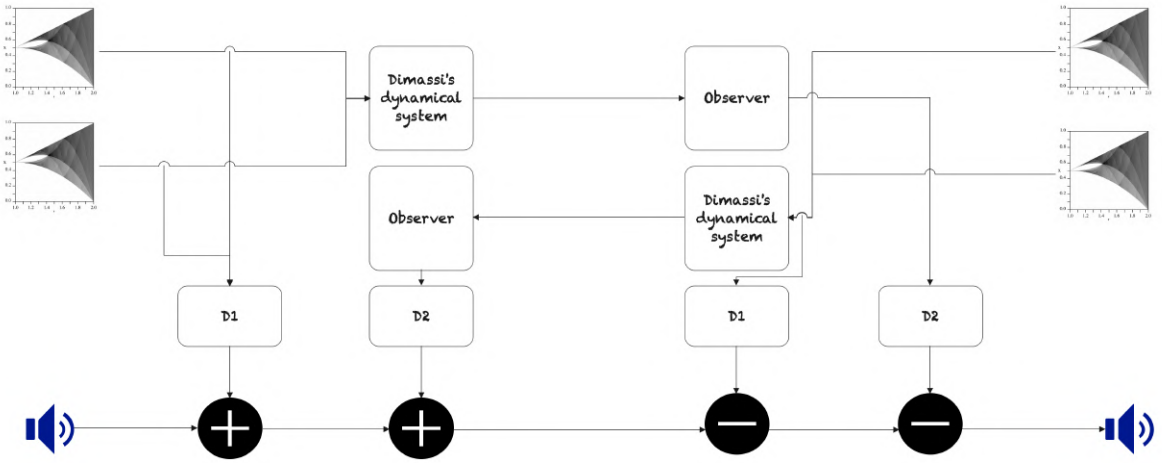


Figure 3.21: The Skew tent maps audio cipher

3.3.4 The Skew tent map audio cryptosystem

The cryptosystem for the secure transmission of audio messages that I have designed is shown in figure 3.21. At the emitter level, we have two skew tent maps, STM_1 and STM_2. The skew tent map used are the ones presented in [19] whose general form is given by 3.53.

$$x_{n+1} = \begin{cases} \frac{2}{\alpha+1}x_n + \frac{1-\alpha}{\alpha+1}, & -1 < x_n < \alpha \\ \frac{2}{\alpha-1}x_n - \frac{\alpha+1}{\alpha-1}, & \alpha \leq x_n < 1 \end{cases} \tag{3.53}$$

where α is a parameter such that $\{\alpha, x_0\} \subset (-1,1)$.

The skew tent maps are used according to a decision algorithm D_1 for masking the plaintext, the result of this masking being a signal w. The two skew tent maps are also used as unknown inputs of Dimassi's dynamical system, the output of which is sent to the receiver. The emitter also receives from the receiver a signal which, once entered in the modified adaptive sliding mode unknown inputs observer, makes it possible to recover two skew tent maps located at the level of the receiver. These skew tent maps are then used according to a decision algorithm D_2 for masking the signal w. At the level of the receiver, we first use an observer to recover the skew tent maps from the emitter and use them according to D_1 to subtract them from the ciphertext. The receiver's skew tent maps are also used according to D_2 to subtract them from the ciphertext and thus recover the plaintext. As with the previous stream cipher, we impose the necessary delays to compensate for the delays caused by the observers.

The decision algorithms D_1 and D_2 are functions given by 3.54.

$$\begin{aligned} D_1 &= s_1 STM_1 + s_2 STM_2 \\ D_2 &= s_1 STM_3 + s_2 STM_4 \end{aligned} \tag{3.54}$$

where s_1 and s_2 are bit streams generated by LFSRs. It is a symmetric cryptosystem. The encryption key is given by 3.55.

$$key = (p_{i1}, p_{j2}, STM_{10}, STM_{20}, \alpha_1, \alpha_2) \tag{3.55}$$

where p_{i1} is the key of the LFSR generating s_1, p_{j2} is the key of the LFSR generating s_2, STM_{i0} is the initial condition of the skew tent map STM_i and α_i is the parameter of the skew tent map STM_i. As for the previous stream cipher, the skew tent maps of the emitter and the skew tent maps of the receiver are the same (same parameters and same initial conditions), so STM_1 is STM_3 and STM_2 is STM_4, the only difference is that because of the noise, the chaotic signals will be different.

3.4 Conclusion

I have analyzed in this chapter two observers for the synchronization of chaotic systems. An adaptive unknown inputs observer and an adaptive sliding mode unknown inputs observer, both constructed in [12]. Each of these observers is used in this work to estimate a certain category of signals. I also made some changes to the observers to improve their performance. These modifications can be considered as general techniques for improving observers for the estimation of unknown signals whose value generation frequency is known. I briefly presented two symmetric chaos-based cryptosystems. One of them is a stream cipher and the other is a system for transmitting audio signals. I also presented a technique for improving the frequency characteristics of chaotic systems, which are for the most part bad for cryptography because of their poor frequency properties. The technique that I have presented consists of noising in a controlled way the derivatives of the states of the chaotic system, so as to obtain a system with good frequency characteristics. I applied this technique on the Rossler system and on the Lorenz system, for which I was able to construct a set of new attractors according to the way in which I added noise to the derivatives of the states. The logical continuation of this work is to detail the two cryptosystems, to propose a material realization, and to make cryptanalysis of them. I am working actually on the design of circuits for the

modified Lorenz attractors. I am also calculating the entropy, Lyapunov exponents and auto-correlation of the modified attractors. Then, I will work on the modification of strange attractors in order to obtain better attractors for systems other than the Rossler system or the Lorenz system.

CHAPTER 4

CHAOS SYNCHRONIZATION USING HIGHER ORDER SLIDING MODE OBSERVERS FOR SYSTEMS WITH TIME DELAY

4.1 Introduction

The objective of this chapter is to analyze Higher-order sliding mode observers which are observers often used for the synchronization of chaos in the presence of delay in the transmission channel. I will start with the predictor-based super twisting second-order sliding mode observer constructed in [18]. I will try to improve this observer by adding an estimation of unknown input. I will also use a technique present in some articles such as [27, 28] which consists of replacing the sign function of sliding mode observers with a fuzzy inference system in order to eliminate the chattering effect due to the discontinuity of the sign function. I will then try to introduce the idea of an adaptive order for the Higher-order sliding mode observers, through the example of the super-twisting observer. This order will vary according to an adaptation law in order to minimize an optimization criterion.

4.2 Predictor-based super-twisting second-order sliding mode observer

I present in this section a predictor-based super twisting higher-order sliding mode observer constructed in [18]. It is an observer for systems with delayed output. I will introduce two modifications of this observer. First, I will try to obtain the estimate of an unknown input, then I will replace the sign functions with fuzzy inference systems in order to eliminate the chattering effect due to the discontinuity of the sign function.

4.2.1 The original observer

The predictor-based super twisting second order sliding mode observer is an observer for systems of the form 4.1.

$$\begin{cases} \dot{x}(t) = f(x(t), d(t)) \\ y^{\tau}(t) = h(x(t-\tau)) \end{cases} \tag{4.1}$$

where $\tau > 0$ is a constant and known time delay and $y^{\tau}(t) := y(t-\tau)$.

If 4.1 is uniformly locally observable, then there exists a local state coordinate transformation $z(t) = \Gamma(x(t))$ which transforms 4.1 into the triangular observable form 4.2.

$$\begin{cases} \dot{z}(t) = Az(t) + \tilde{\Phi}(z(t)) + B\eta(t)(t) \\ y^{\tau} = Cz^{\tau}(t) \end{cases} \tag{4.2}$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ . & . & . & . & & \\ . & . & . & . & \dots & 0 \\ 0 & 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 \end{bmatrix}, \; B = \begin{bmatrix} 0 & 0 & 0 & \dots & 1 \end{bmatrix}^T, \; C = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \end{bmatrix}$$

and

$$\tilde{\Phi}(z) = \begin{bmatrix} \Phi_1(z_1) \\ \Phi_2(z_1, z_2) \\ . \\ . \\ 0 \end{bmatrix}$$

and $\eta(t) = \Phi_n(z_1, z_2, ..., z_n) + d(t)$ is the total disturbance, where $d(t)$ is an unknown external disturbance and $\Phi_n(z_1, z_2, ..., z_n)$ represents the unknown system uncertainties.

We can rewrite 4.2 in the more explicit form 4.3.

$$\begin{cases} \dot{z}_1(t) = z_2(t) + \Phi_1(z_1) \\ \dot{z}_2(t) = z_3(t) + \Phi_2(z_1, z_2) \\ \dots \\ \dot{z}_{n-1}(t) = z_n(t) + \Phi_{n-1}(z_1, z_2, ..., z_{n-1}) \\ \dot{z}_n(t) = \eta(t) \\ y^\tau(t) = z_1(t-\tau) \end{cases}$$

The super-twisting sliding mode observer for 4.3 is given by 4.3 [18].

$$\begin{aligned} \dot{\hat{z}}_1^\tau &= \tilde{z}_2^\tau + \lambda_1 \sqrt{|\epsilon_1^\tau|}\, sign(\epsilon_1^\tau) + \Phi_1(z_1^\tau) \\ \dot{\tilde{z}}_2^\tau &= \alpha_1 sign(\epsilon_1^\tau) \\ \dot{\hat{z}}_2^\tau &= E_1[\tilde{z}_3^\tau + \lambda_2 \sqrt{|\epsilon_2^\tau|}\, sign(\epsilon_2^\tau) + \Phi_2(z_1^\tau, \tilde{z}_2^\tau)] \\ \dot{\tilde{z}}_3^\tau &= E_1[\alpha_2 sign(\epsilon_2^\tau)] \\ \dot{\hat{z}}_3^\tau &= E_2[\tilde{z}_4^\tau + \lambda_3 \sqrt{|\epsilon_3^\tau|}\, sign(\epsilon_3^\tau) + \Phi_3(z_1^\tau, \tilde{z}_2^\tau, \tilde{z}_3^\tau)] \\ &\dots \\ \dot{\tilde{z}}_{n-1}^\tau &= E_{n-3}[\alpha_{n-2} sign(\epsilon_{n-2}^\tau)] \\ \dot{\hat{z}}_{n-1}^\tau &= E_{n-2}[\tilde{z}_n^\tau + \lambda_{n-1} \sqrt{|\epsilon_{n-1}^\tau|}\, sign(\epsilon_{n-1}^\tau) + \Phi_{n-1}(z_1^\tau, \tilde{z}_2^\tau, ..., \tilde{z}_{n-1}^\tau)] \\ \dot{\tilde{z}}_n^\tau &= E_{n-2}[\alpha_{n-1} sign(\epsilon_{n-1}^\tau)] \\ \dot{\hat{z}}_n^\tau &= E_{n-1}[\tilde{\theta}^\tau + \lambda_n \sqrt{|\epsilon_n^\tau|}\, sign(\epsilon_n^\tau)] \\ \dot{\tilde{\theta}}^\tau &= E_{n-1}[\alpha_n sign(\epsilon_n^\tau)] \end{aligned} \tag{4.3}$$

where :

$$\begin{aligned} \epsilon_i^\tau &= \tilde{z}_i^\tau - \hat{z}_i^\tau \\ \tilde{z}_1^\tau = z_1^\tau = y^\tau &\Longrightarrow \epsilon_1^\tau = e_1^\tau = z_1^\tau - \hat{z}_i^\tau \end{aligned}$$

and $\hat{z}_i^\tau$ and $\tilde{z}_i^\tau$ are the delayed estimated state and the delayed internal state of the observer, respectively. The function E_i is given by 4.4.

$$E_i = \begin{cases} 1 & if \quad |\epsilon_j^\tau| \le \epsilon, \quad \forall j \le i \\ 0 & else \end{cases} \tag{4.4}$$

where ϵ is a small positive constant.

The parameters α_i are the observer gains and the parameters $\lambda_i > 0$ are the correction factors that are here to ensure the convergence of the observer.

We have the following result about the convergence of 4.3 [18]: If the delayed states of 4.3 are uniformly bounded in a compact domain $D \subset \mathbf{R}^n$, i.e. $|z_i^\tau| \leq \sigma_i, \quad i = 1,2,...,n)$, and

$$|\Phi_i(z_j^\tau, j = 1,2,...,i)| \leq L_i, \quad i = 1,2,...,n-1$$

and

$$|\frac{d\eta}{dt}| \leq L_n$$

then, for any initial conditions $\tilde{z}_i$, $\hat{z}_i$, $i = 1,...,n$ and $\tilde{\theta}^\tau$ there exist positive constants λ_j and α_j for $j = 1,2...n$, satisfying the conditions 4.5.

$$\begin{aligned}
&\alpha_j > (L_{j+1} + \alpha_{j+2}) \\
&\lambda_j > \sqrt{4(L_{j+1} + \sigma_{j+2})\frac{\alpha_j + (L_{j+1} + \sigma_{j+2})}{\alpha_j - (L_{j+1} + \sigma_{j+2})}}, \quad j = 1,,...,n-1 \\
&\alpha_n > L_n \\
&\lambda_n > \sqrt{4L_n\frac{\alpha_n + L_n}{\alpha_n - L_n}}
\end{aligned} \tag{4.5}$$

such that the estimated delayed states $\hat{z}_j^\tau$ converge to the delayed states z_j^τ in finite time. In addition $\tilde{\theta}^\tau$ converges to the delayed disturbance η^τ. More explicitly, $e_n^\tau(t)$ and $\dot{e}_n^\tau(t)$ converges to zero in a finite time T_n satisfying 4.6.

$$T_n \leq \Sigma\frac{|\dot{e}_{ni}^\tau|}{|\alpha_n - L_n|} \tag{4.6}$$

It follows that a delayed estimate of the delayed total disturbance can be obtained in finite time T_n : $\hat{\eta}^\tau(t) = \tilde{\theta}^\tau(t)$. In other words, there exists a finite time T_n such that 4.7 holds.

$$\begin{aligned}
&||z^\tau(t) - \hat{z}^\tau(t)|| = 0, \quad t \geq T_n + \tau \\
&|\eta^\tau(t) - \hat{\eta}^\tau(t)| = 0, \quad t \geq T_n + \tau
\end{aligned} \tag{4.7}$$

The super-twisting sliding mode observer 4.3 estimates the delayed states. We use the predictor 4.8 in cascade with 4.3 to estimate the actual states.

$$\begin{cases}
\dot{\gamma} = A\gamma + \tilde{\Phi}(z^p) + B\hat{\eta}^\tau \\
z^p = e^{A\tau}\hat{z}^\tau + \gamma - e^{A\tau}\gamma^\tau
\end{cases} \tag{4.8}$$

where $\gamma \in \mathbf{R}^n$ is the internal state of the predictor.

If $\tilde{\Phi}$ is globally Lipshitz with respect to z with a Lipshitz constant α_z, and η satisfies the slowly time varying condition, i.e $|\eta - \eta^\tau| \leq \epsilon_\eta$, $\forall t$, $\forall \tau \in [0, \tau^*]$ where τ^* is an upper bound of the delay and ϵ_η a positive small real constant, then the prediction error $e^p = z - z^p$ converges to a ball of radius $K\epsilon_\eta$, for all $t \geq T_n + \tau$, where $K = ||e^{A\tau}||\tau e^{a\tau}$ and $a = \alpha_z ||e^{A\tau}||$.

4.2.2 Example : the classical Rossler system

The classical Rossler system in its triangular form is given by 4.9.

$$\begin{cases} \dot{z}_1(t) = z_2(t) \\ \dot{z}_2(t) = z_3(t) \\ \dot{z}_3(t) = \Phi_3(z_1, z_2, z_3) + d(t) \\ y^\tau(t) = z_1(t - \tau) \end{cases} \tag{4.9}$$

where $d(t) = 0.1sin(t)$ and

$$\Phi_3(z_1, z_2, z_3) = -z_2(t) + az_3(t) - (z_1(t) + z_3(t) - az_2(t))(c - z_2(t) + az_1(t)) - b$$

where $a = 0.2$, $b = 0.2$ and $c = 5.7$. It is a system of the form 4.2 where $\eta(t) = \Phi_3(z_1, z_2, z_3) + d(t)$ and

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \quad C = [1 \;\; 0 \;\; 0], B = [0 \;\; 0 \;\; 1]^T, \quad \tilde{\Phi}(z) = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

The attractor of this system is shown in figure 4.1. The super-twisting sliding mode observer for 4.9 is given by 4.10.

$$\begin{aligned} \dot{\hat{z}}_1^\tau &= \tilde{z}_2^\tau + \lambda_1 \sqrt{|\epsilon_1^\tau|}\, sign(\epsilon_1^\tau) \\ \dot{\tilde{z}}_2^\tau &= \alpha_1 sign(\epsilon_1^\tau) \\ \dot{\hat{z}}_2^\tau &= E_1[\tilde{z}_3^\tau + \lambda_2 \sqrt{|\epsilon_2^\tau|}\, sign(\epsilon_2^\tau)] \\ \dot{\tilde{z}}_3^\tau &= E_1[\alpha_2 sign(\epsilon_2^\tau)] \\ \dot{\hat{z}}_3^\tau &= E_2[\tilde{\theta}^\tau + \lambda_3 \sqrt{|\epsilon_3^\tau|}\, sign(\epsilon_3^\tau)] \\ \dot{\tilde{\theta}}^\tau &= E_2[\alpha_3 sign(\epsilon_3^\tau)] \end{aligned} \tag{4.10}$$

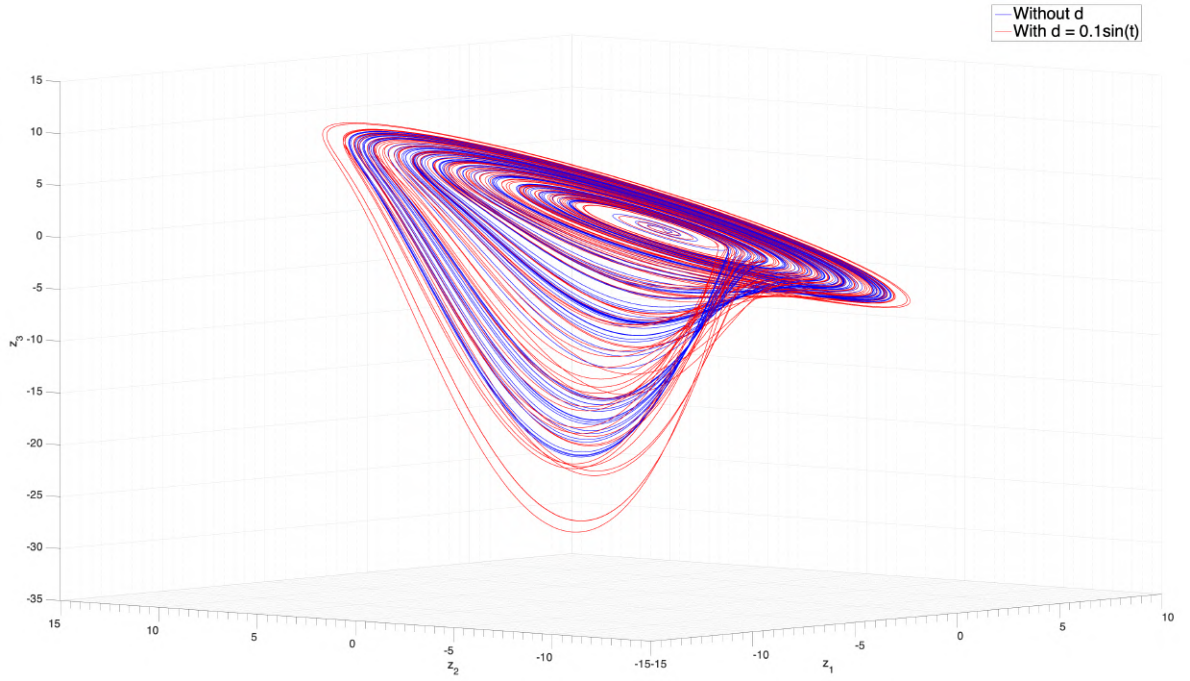


Figure 4.1: The attractor of the Rossler triangular system

where we take the gains 4.11.

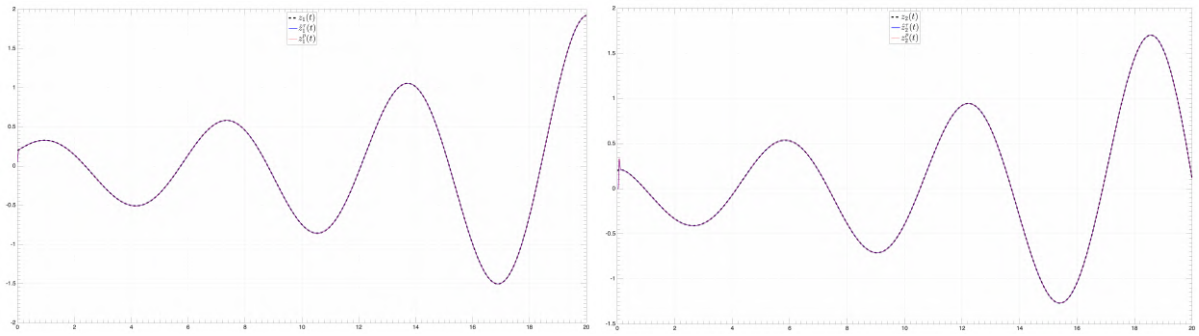
$$\begin{aligned} &\alpha_1 = 30; \quad \alpha_2 = 30; \quad \alpha_3 = 30; \\ &\lambda_1 = 15; \quad \lambda_2 = 15; \quad \lambda_2 = 15. \end{aligned} \tag{4.11}$$

The predictor is given by 4.12.

$$\begin{aligned} &\dot{\gamma}(t) = A\gamma(t) + B\hat{\eta}^{\tau}(t) \\ &z^{p}(t) = e^{A\tau}\hat{z}^{\tau}(t) + \gamma(t) - e^{A\tau}\gamma^{\tau}(t) \end{aligned} \tag{4.12}$$

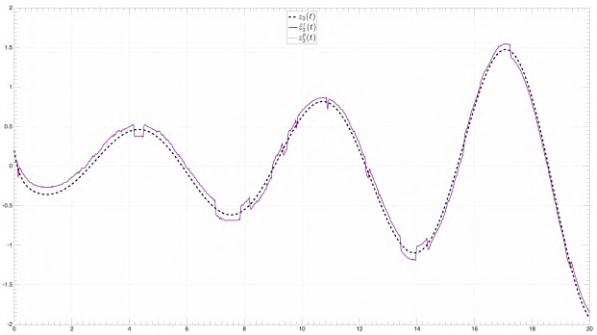
The initial conditions are taken as $z_1(0) = 0.2$, $z_2(0) = 0.2$, $z_3(0) = 0.2$, $\hat{z}_1^{\tau}(\tau) = \hat{z}_3^{\tau}(\tau) = \tilde{z}_3^{\tau}(\tau) = 0.05$, $\hat{z}_2^{\tau}(\tau) = \tilde{z}_2^{\tau}(\tau) = 0$, $\hat{\gamma}_1(0) = \hat{\gamma}_3(0) = 0.05$ and $\hat{\gamma}_2(0) = 0$. The value of ϵ is fixed to 0.0025. I used a time step of $10^{-5}s$[1]. The results of the simulations for $\tau = 0$ and $\tau = 0.5s$ are shown in figures 4.2 and 4.3 respectively.

[1]When I used the "auto" time step of SIMULINK, I found bad results. The reason is that with greater time step, there are more numerical errors. So one needs to find the good computation frequency when implementing an observer.

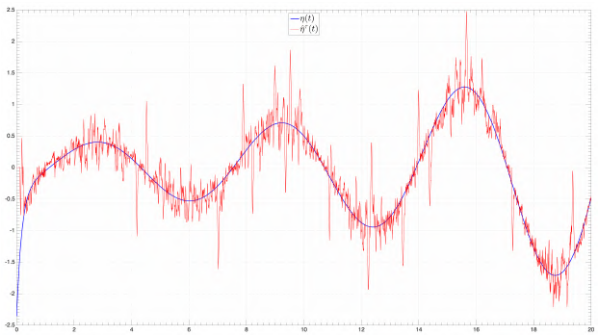


(a) The estimation of z_1

(b) The estimation of z_2

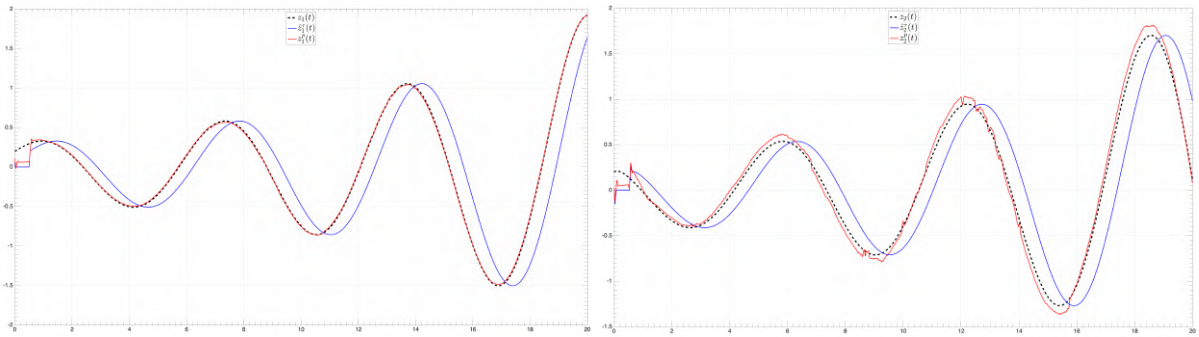


(c) The estimation of z_3



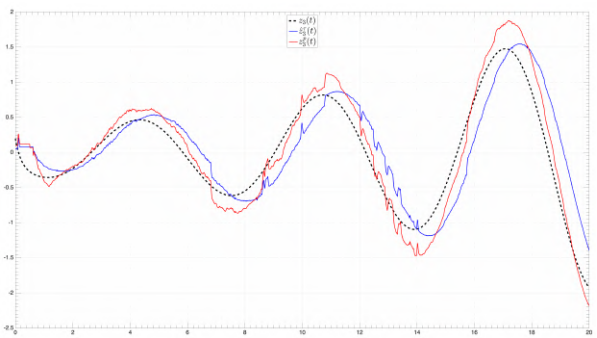
(d) The estimation of η

Figure 4.2: The simulations for $\tau = 0$

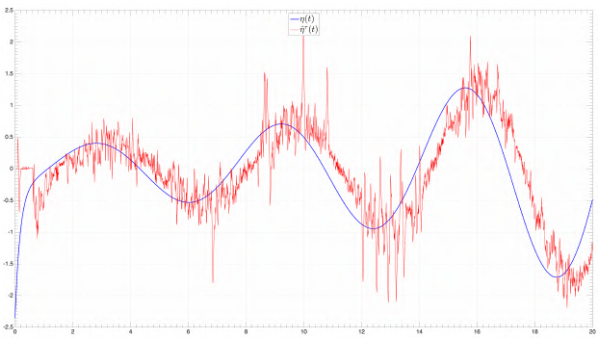


(a) The estimation of z_1

(b) The estimation of z_2



(c) The estimation of z_3



(d) The estimation of η

Figure 4.3: The simulations for $\tau = 0.5s$

4.2.3 The modified predictor-based super-twisting second order sliding mode observer

My first modification is to replace the sign functions with fuzzy inference systems. This technique is used in some articles such as [27, 28]. My objective behind this is to attenuate the chattering effect due to the use of the discontinuous sign function. The fuzzy super-twisting observer is given by 4.13. The predictor does not change.

$$
\begin{aligned}
&\dot{\hat{z}}_1^\tau = \tilde{z}_2^\tau + \lambda_1 \sqrt{|\epsilon_1^\tau|}\,\psi_1(\epsilon_1^\tau) + \Phi_1(z_1^\tau) \\
&\dot{\hat{z}}_2^\tau = \alpha_1 \psi_1(\epsilon_1^\tau) \\
&\dot{\hat{z}}_2^\tau = E_1[\tilde{z}_3^\tau + \lambda_2 \sqrt{|\epsilon_2^\tau|}\,\psi_2(\epsilon_2^\tau) + \Phi_2(z_1^\tau, \tilde{z}_2^\tau)] \\
&\dot{\hat{z}}_3^\tau = E_1[\alpha_2 \psi_2(\epsilon_2^\tau)] \\
&\dot{\hat{z}}_3^\tau = E_2[\tilde{z}_4^\tau + \lambda_3 \sqrt{|\epsilon_3^\tau|}\,\psi_3(\epsilon_3^\tau) + \Phi_3(z_1^\tau, \tilde{z}_2^\tau, \tilde{z}_3^\tau)] \\
&\ldots \\
&\dot{\hat{z}}_{n-1}^\tau = E_{n-3}[\alpha_{n-2} \psi_{n-2}(\epsilon_{n-2}^\tau)] \\
&\dot{\hat{z}}_{n-1}^\tau = E_{n-2}[\tilde{z}_n^\tau + \lambda_{n-1} \sqrt{|\epsilon_{n-1}^\tau|}\,\psi_{n-1}(\epsilon_{n-1}^\tau) + \Phi_{n-1}(z_1^\tau, \tilde{z}_2^\tau, ..., \tilde{z}_{n-1}^\tau)] \\
&\dot{\hat{z}}_n^\tau = E_{n-2}[\alpha_{n-1} \psi_{n-1}(\epsilon_{n-1}^\tau)] \\
&\dot{\hat{z}}_n^\tau = E_{n-1}[\tilde{\theta}^\tau + \lambda_n \sqrt{|\epsilon_n^\tau|}\,\psi_n(\epsilon_n^\tau)] \\
&\dot{\hat{\theta}}^\tau = E_{n-1}[\alpha_n \psi_n(\epsilon_n^\tau)]
\end{aligned}
\tag{4.13}
$$

where ψ_1, ψ_2, ..., ψ_n are fuzzy inference systems. To design the fuzzy inference systems, we study the errors ϵ_i to see their bounds (for example : big negative error if the error is between a_1 and a_2, etc ...), and then we create corresponding zones for the output ψ (for example : $\psi = 1$ if the error is positive big, etc ...). I designed one fuzzy inference system for each sign function. The three fuzzy inference systems for the Rossler system are shown in figure 4.4 where the i^{th} row corresponds to ψ_i. I used *If ... Then* rules :

- If ϵ_i^τ is **NBB** then ψ_i is **NBB**.
- If ϵ_i^τ is **NB** then ψ_i is **NB**.
- If ϵ_i^τ is **NS** then ψ_i is **NS**.
- If ϵ_i^τ is **ZR** then ψ_i is **ZR**.
- If ϵ_i^τ is **PS** then ψ_i is **PS**.
- If ϵ_i^τ is **PB** then ψ_i is **PB**.
- If ϵ_i^τ is **PBB** then ψ_i is **PBB**.

for $i = 2,3$. For $i = 1$, the rules are :

- If ϵ_1^τ is **NB** then ψ_1 is **NB**.
- If ϵ_1^τ is **NS** then ψ_1 is **NS**.
- If ϵ_1^τ is **ZR** then ψ_1 is **ZR**.
- If ϵ_1^τ is **PS** then ψ_1 is **PS**.
- If ϵ_1^τ is **PB** then ψ_1 is **PB**.

j where :

- **NBB** : Negative Big Big.
- **NB** : Negative Big.
- **NS** : Negative Small.
- **ZR** : Zero.
- **PS** : Positive Small.
- **PBB** : Positive Big Big.

The results of the simulations are shown in figures 4.5 and 4.6. As we can see, with fuzzy inference systems there is no chattering effect. In figure 4.5, if we continue the simulations over $20s$, the fuzzy based observer for $\tau = 0$ does not converge, we need to change the value of ϵ to 0.025 for example to remedy this. I think that the reason of this is that the error ϵ_3^τ becomes greater than the bounds used for the design of the fuzzy inference system (I saw that in the simulations). Even later, when I will design the second modification, the Rossler system will not always react well (after $20s$, the unknown input estimation will become very bad). I used for the second modification another system which is simpler and the equations worked well on it. It is yet not clear why the Rossler system reacts badly.

The second modification is for the estimation of unknown input. I will consider now d not as an external disturbance but as an external unknown input that I want to estimate, and Φ_n as a known function (which is the case for example for the classical Rossler system). This technique consists of treating the unknown input as a system state. It can be generalized to all systems that are written in a triangular form. A new "fictitious" state is therefore created. It is fictitious in the sense that its derivative is known and does not depend on any state, given that it is we ourselves who generated the "fictitious"

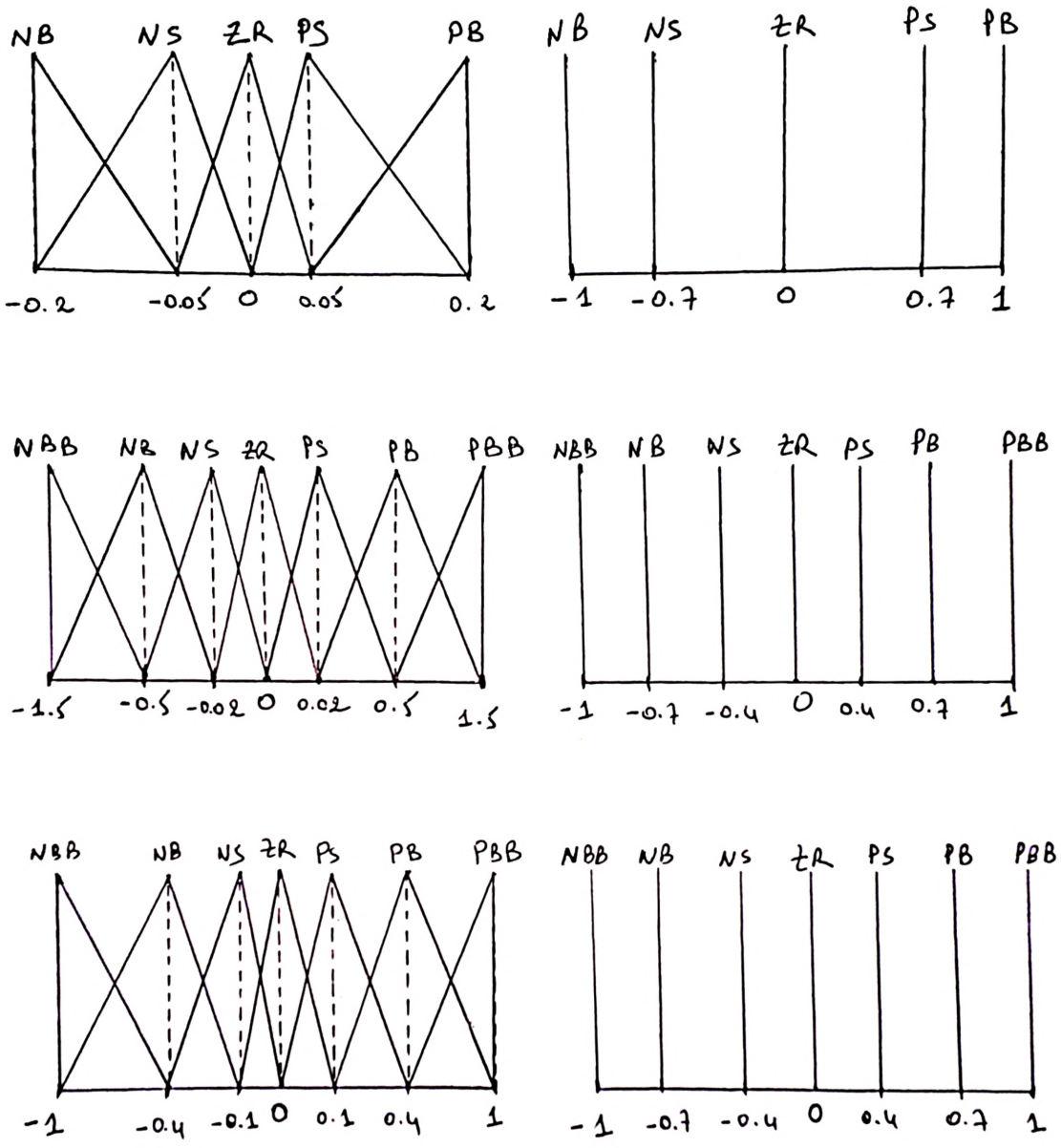
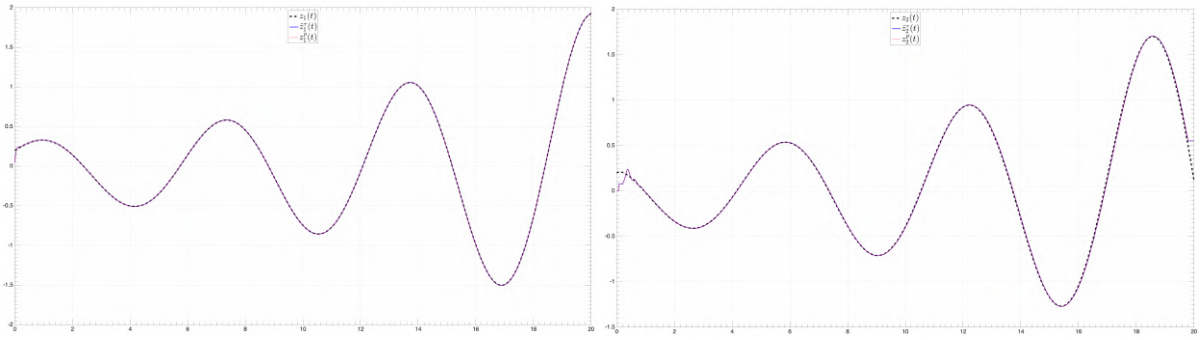
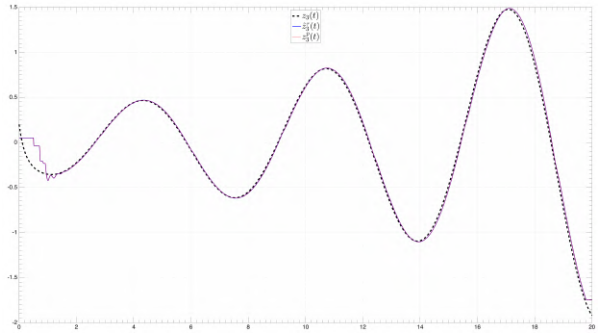


Figure 4.4: The fuzzy inference systems

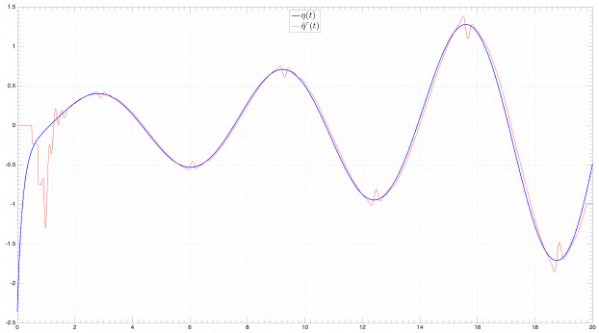


(a) The estimation of z_1

(b) The estimation of z_2

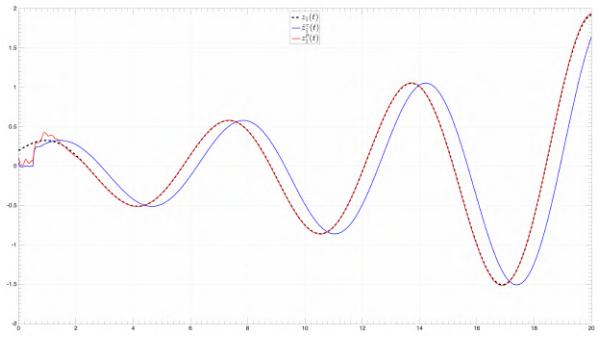


(c) The estimation of z_3

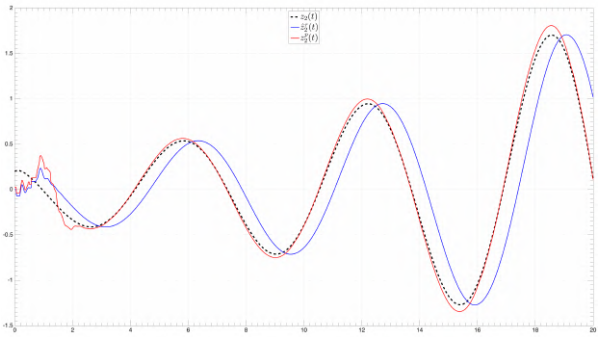


(d) The estimation of η

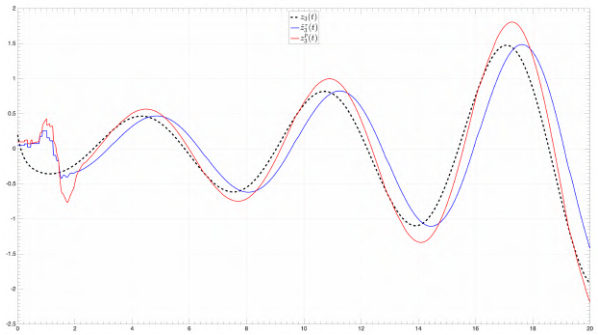
Figure 4.5: The simulations for $\tau = 0$ using fuzzy inference systems



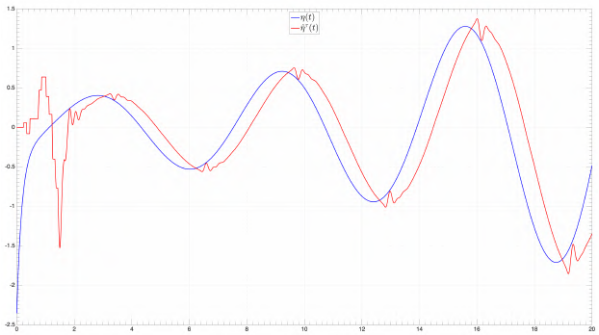
(a) The estimation of z_1



(b) The estimation of z_2



(c) The estimation of z_3



(d) The estimation of η

Figure 4.6: The simulations for $\tau = 0.5s$ using fuzzy inference systems

state. The new triangular representation of the dynamical system is given by 4.14.

$$\begin{cases} \dot{z}_1(t) = z_2(t) + \Phi_1(z_1) \\ \dot{z}_2(t) = z_3(t) + \Phi_2(z_1, z_2) \\ \dots \\ \dot{z}_{n-1}(t) = z_n(t) + \Phi_{n-1}(z_1, z_2, \dots, z_{n-1}) \\ \dot{z}_n(t) = d + \Phi_n(z_1, z_2, \dots, z_n) \\ \dot{d}(t) = m \\ y^\tau(t) = z_1(t-\tau) \end{cases} \tag{4.14}$$

where m is the derivative of the state that we generate.

The structure of the new observer is given by 4.15.

$$\begin{aligned} &\dot{\hat{z}}_1^\tau = \tilde{z}_2^\tau + \lambda_1 \sqrt{|\epsilon_1^\tau|}\, \psi_1(\epsilon_1^\tau) + \Phi_1(z_1^\tau) \\ &\dot{\tilde{z}}_2^\tau = \alpha_1 \psi_1(\epsilon_1^\tau) \\ &\dot{\hat{z}}_2^\tau = E_1[\tilde{z}_3^\tau + \lambda_2 \sqrt{|\epsilon_2^\tau|}\, \psi_2(\epsilon_2^\tau) + \Phi_2(z_1^\tau, \tilde{z}_2^\tau)] \\ &\dot{\tilde{z}}_3^\tau = E_1[\alpha_2 \psi_2(\epsilon_2^\tau)] \\ &\dot{\hat{z}}_3^\tau = E_2[\tilde{z}_4^\tau + \lambda_3 \sqrt{|\epsilon_3^\tau|}\, \psi_3(\epsilon_3^\tau) + \Phi_3(z_1^\tau, \tilde{z}_2^\tau, \tilde{z}_3^\tau)] \\ &\dots \\ &\dot{\tilde{z}}_{n-1}^\tau = E_{n-3}[\alpha_{n-2} \psi_{n-2}(\epsilon_{n-2}^\tau)] \\ &\dot{\hat{z}}_{n-1}^\tau = E_{n-2}[\tilde{z}_n^\tau + \lambda_{n-1} \sqrt{|\epsilon_{n-1}^\tau|}\, \psi_{n-1}(\epsilon_{n-1}^\tau) + \Phi_{n-1}(z_1^\tau, \tilde{z}_2^\tau, \dots, \tilde{z}_{n-1}^\tau)] \\ &\dot{\tilde{z}}_n \tau = E_{n-2}[\alpha_{n-1} \psi_{n-1}(\epsilon_{n-1}^\tau)] \\ &\dot{\hat{z}}_n^\tau = E_{n-1}[\tilde{d}^\tau + \lambda_n \sqrt{|\epsilon_n^\tau|}\, \psi_n(\epsilon_n^\tau) + \Phi_n(z_1^\tau, \tilde{z}_2^\tau, \dots, \tilde{z}_{n-1}^\tau, \tilde{z}_n^\tau)] \\ &\dot{\tilde{d}} \tau = E_{n-1}[\alpha_n \psi_n(\epsilon_n^\tau)] \\ &\dot{\hat{d}}^\tau = E_n[\tilde{\theta}^\tau + \lambda_{n+1} \sqrt{|\epsilon_{n+1}^\tau|}\, \psi_{n+1}(\epsilon_{n+1}^\tau)] \\ &\dot{\tilde{\theta}}^\tau = E_n[\alpha_{n+1} \psi_{n+1}(\epsilon_{n+1}^\tau)] \end{aligned} \tag{4.15}$$

where $\epsilon_{n+1} = \tilde{d} - \hat{d}$ and E_n is defined as before. For the predictor, we just need to add an additional state and use the corresponding matrices.

I applied this modified observer on the system 4.16.

$$\begin{cases} z_1 = z_2 \\ z_2 = z_3 \\ z_3 = -z_3 + d \end{cases} \tag{4.16}$$

The results of the simulations[2] are shown in figure 4.7[3].

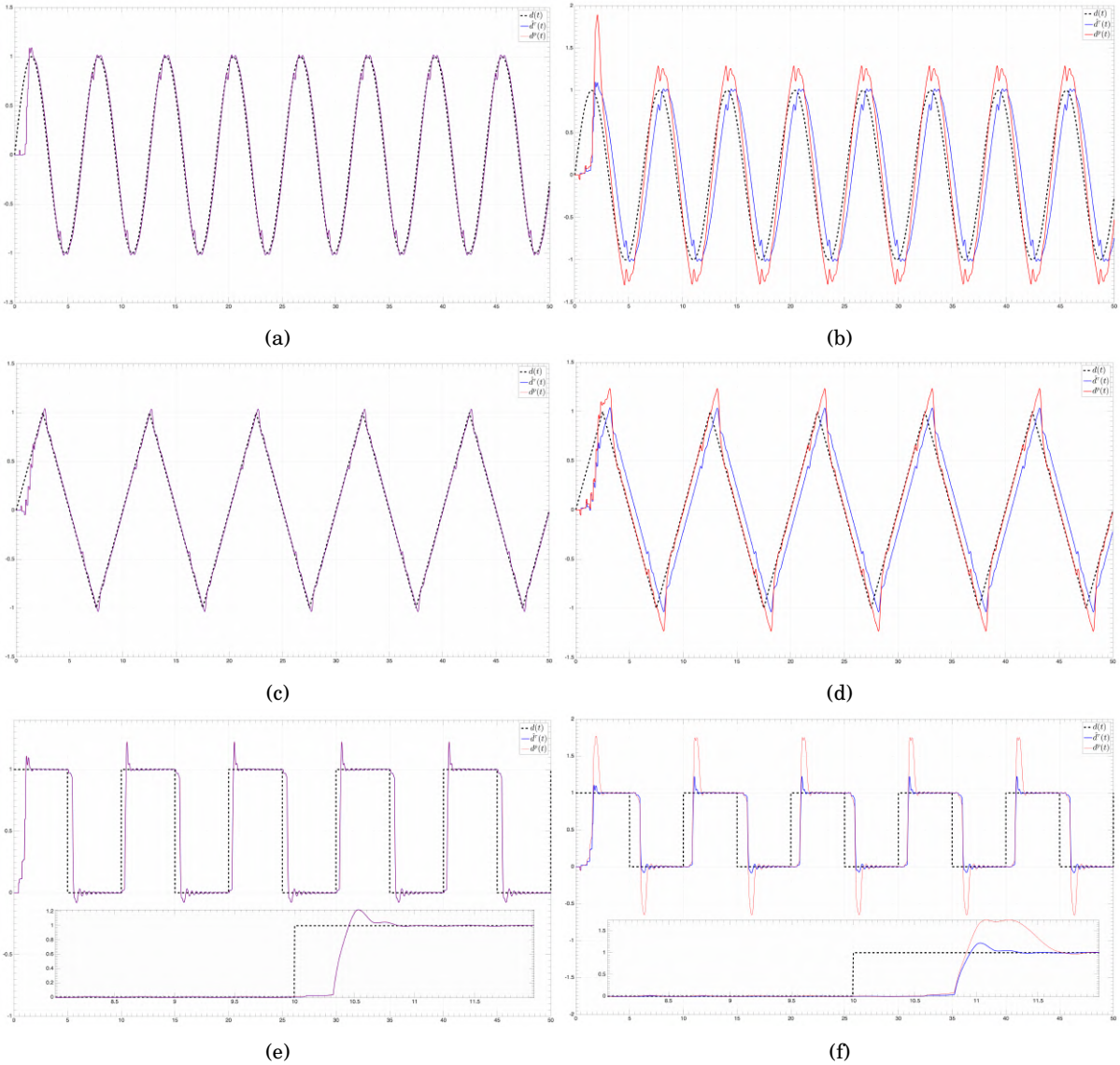


Figure 4.7: The simulations for $\tau = 0$ (left) and $\tau = 0.5s$ (right) of unknown input estimation

[2]More simulations have to be done to see if the observer is robust to different noises (noise at each state, noise in the output, etc ...).

[3]The predictor does not work well for the square input. The reason is that the second derivative of the square input is very high.

I did the simulations with the "auto" time step of SIMULINK to show that we do not need a high computation frequency to achieve good estimations when we use fuzzy inference systems with the super twisting observer For the fourth sign function, I used ψ_3. I used the value $\epsilon = 0.025$ and I took $\lambda_i = 15$ and $\alpha_i = 30$ for $i = 1,2,3,4$. I did also the simulations using sign functions with "auto" time stem (left) and a time step of $10^{-5}s$ (right). The results are shown in figures 4.8 and 4.9. As we can see the estimation becomes much noisier and needs a higher frequency computation when we use sign functions instead of fuzzy inference systems.

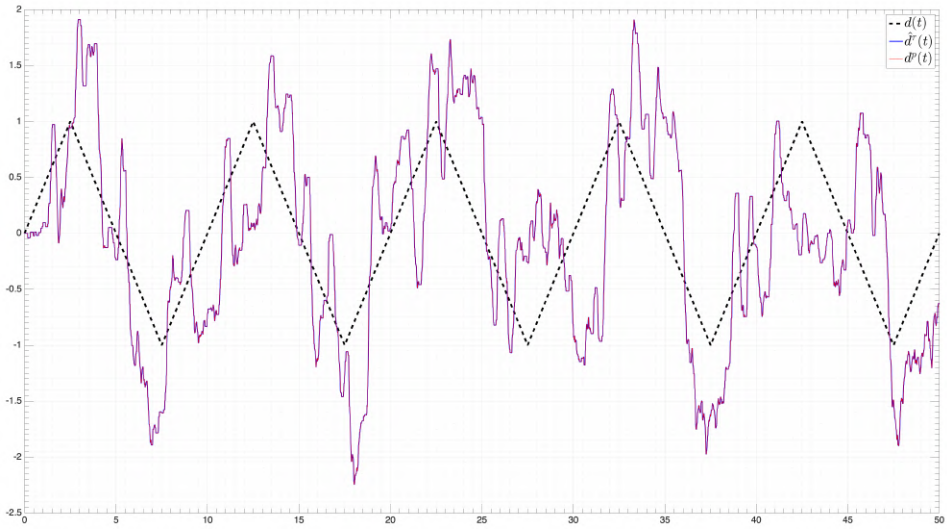


Figure 4.8: Unknown input estimation for $\tau = 0$ with "auto" time step

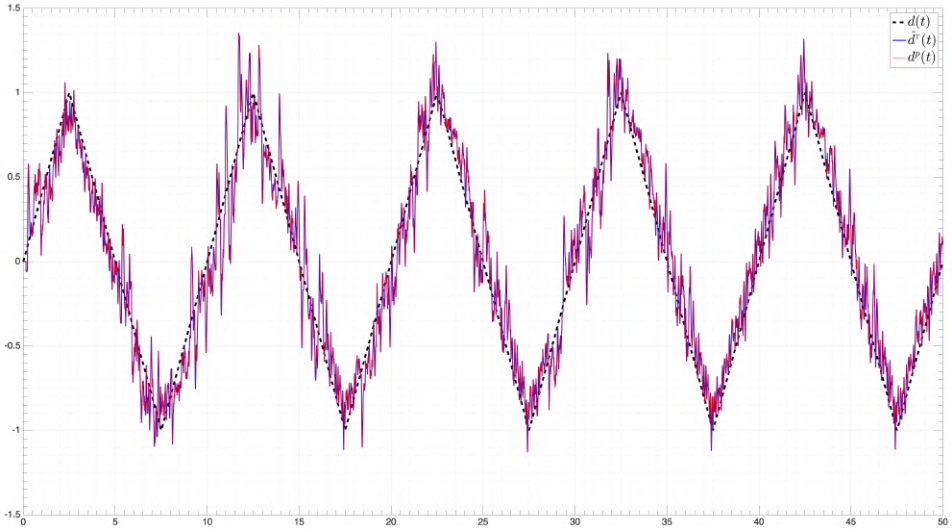


Figure 4.9: Unknown input estimation for $\tau = 0$ with $10^{-5}s$ time step

4.3 Adaptive-order fuzzy sliding mode observer

In this section I present an improvement of the super-twisting fuzzy sliding mode observer. The new structure of the observer is given by 4.17.

$$
\begin{aligned}
&\dot{\hat{z}}_1^\tau = \tilde{z}_2^\tau + \lambda_1 |\epsilon_1^\tau|^\gamma \psi_1(\epsilon_1^\tau) + \Phi_1(z_1^\tau) \\
&\dot{\hat{z}}_2^\tau = \alpha_1 \psi_1(\epsilon_1^\tau) \\
&\dot{\hat{z}}_2^\tau = E_1[\tilde{z}_3^\tau + \lambda_2 |\epsilon_2^\tau|^\gamma \psi_2(\epsilon_2^\tau) + \Phi_2(z_1^\tau, \tilde{z}_2^\tau)] \\
&\dot{\hat{z}}_3^\tau = E_1[\alpha_2 \psi_2(\epsilon_2^\tau)] \\
&\dot{\hat{z}}_3^\tau = E_2[\tilde{z}_4^\tau + \lambda_3 |\epsilon_3^\tau|^\gamma \psi_3(\epsilon_3^\tau) + \Phi_3(z_1^\tau, \tilde{z}_2^\tau, \tilde{z}_3^\tau)] \\
&\dots \\
&\dot{\hat{z}}_{n-1}^\tau = E_{n-3}[\alpha_{n-2} \psi_{n-2}(\epsilon_{n-2}^\tau)] \\
&\dot{\hat{z}}_{n-1}^\tau = E_{n-2}[\tilde{z}_n^\tau + \lambda_{n-1} |\epsilon_{n-1}^\tau|^\gamma \psi_{n-1}(\epsilon_{n-1}^\tau) + \Phi_{n-1}(z_1^\tau, \tilde{z}_2^\tau, ..., \tilde{z}_{n-1}^\tau)] \\
&\dot{\tilde{z}}_n \tau = E_{n-2}[\alpha_{n-1} \psi_{n-1}(\epsilon_{n-1}^\tau)] \\
&\dot{\hat{z}}_n^\tau = E_{n-1}[\tilde{d}^\tau + \lambda_n |\epsilon_n^\tau|^\gamma \psi_n(\epsilon_n^\tau) + \Phi_n(z_1^\tau, \tilde{z}_2^\tau, ..., \tilde{z}_{n-1}^\tau, \tilde{z}_n^\tau)] \\
&\dot{\tilde{d}} \tau = E_{n-1}[\alpha_n \psi_n(\epsilon_n^\tau)] \\
&\dot{\hat{d}}^\tau = E_n[\tilde{\theta}^\tau + \lambda_{n+1} |\epsilon_{n+1}^\tau|^\gamma \psi_{n+1}(\epsilon_{n+1}^\tau)] \\
&\dot{\tilde{\theta}}^\tau = E_n[\alpha_{n+1} \psi_{n+1}(\epsilon_{n+1}^\tau)]
\end{aligned}
\tag{4.17}
$$

By changing the value of γ, we change the convergence precision of the observer. If our goal is for example to minimize the errors ϵ_j, then we can use the adaptation law 4.18.

$$\dot{\gamma} = -\Gamma \nabla_\gamma \mathscr{L} \tag{4.18}$$

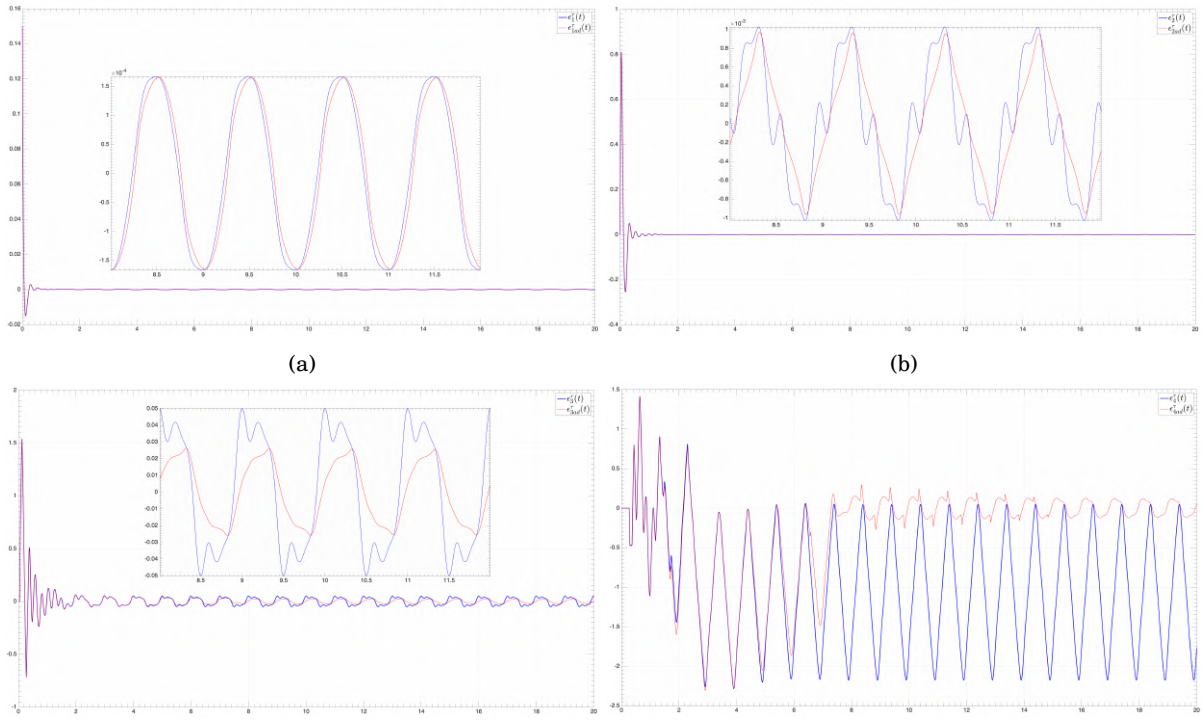
where Γ is the learning rate and $\mathscr{L}$ is a loss function, in our case 4.19.

$$\mathscr{L} = |\epsilon_1^\tau|^2 + |\epsilon_2^\tau|^2 + ... + |\epsilon_n^\tau|^2 \tag{4.19}$$

The results of the simulations for $\Gamma = 0.004$ and $\gamma(0) = 0.5$ are shown in figures 4.10 and 4.11.

4.4 Conclusion

In this chapter, I presented the predictor-based super twisting second-order sliding mode observer built-in [18] for the estimation in the presence of time delay. I did the observer simulations on the Rossler system. I made two changes to the watcher. First, I replaced



(a) (b)

(c) (d)

Figure 4.10: The difference between the original errors ϵ_i^τ and the errors with adaptive order ϵ_{iad}^τ

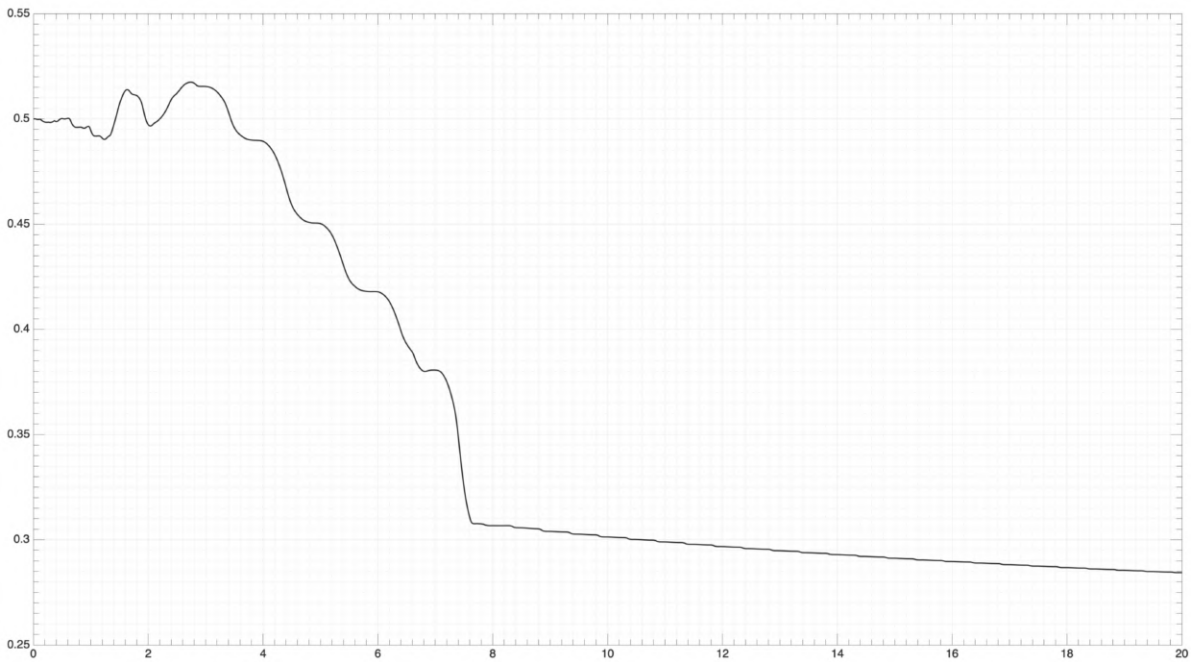


Figure 4.11: The function γ

the sign functions with fuzzy inference systems and simulations have shown that this eliminates the chattering effect. However, some drawbacks have been noted and further study needs to be done. I then introduced a method to consider an external input as a state of the dynamic system. I used this method on a simple dynamic system to make an unknown input estimate using the super twisting observer. The simulations showed that this technique worked. There remains, however, a problem with the Rossler system for which the estimation does not work from $20s$. I ended the chapter by introducing the concept of variable order for higher-order sliding mode observers. This order is now a function that evolves according to an optimization constraint. I applied this technique to the super twisting observer. More simulations need to be done to analyze the robustness of the proposed techniques.

Part II

CHAPTER 5

ACOUSTIC CRYPTANALYSIS COUNTERMEASURE USING IMPROVED LORENZ SYSTEM AND ANFIS

5.1 Introduction

The objective of this chapter is to present a countermeasure against an acoustic cryptanalysis method presented in [16] allowing to break RSA. This cryptanalysis method uses the high-frequency sound emitted by the processor during the decryption operation of RSA in order to find the secret key. The countermeasure I propose is to use a chaotic system in cascade with an ANFIS in order to emit a sound that will mask the sound emitted by RSA decryption. The chaotic system that I am going to use is one of the Lorenz attractors that I obtained with the method of adding noise to the derivative of the state vector. The reason I want to cascade to this chaotic system an ANFIS is that if there is not an ANFIS and Oscar wants to remove the chaotic signal generated, he could use a chaotic synchronization technique and remove the noise I added. In order to prevent Oscar from using chaotic synchronization, I will use an ANFIS as an intermediary between the chaotic circuit and the sound emitted. The objective of ANFIS is to use the output of the chaotic system to generate noise. This noise generation technique is closely dependent on the initial conditions of the chaotic system, which makes it difficult for Oscar to recreate the noise.

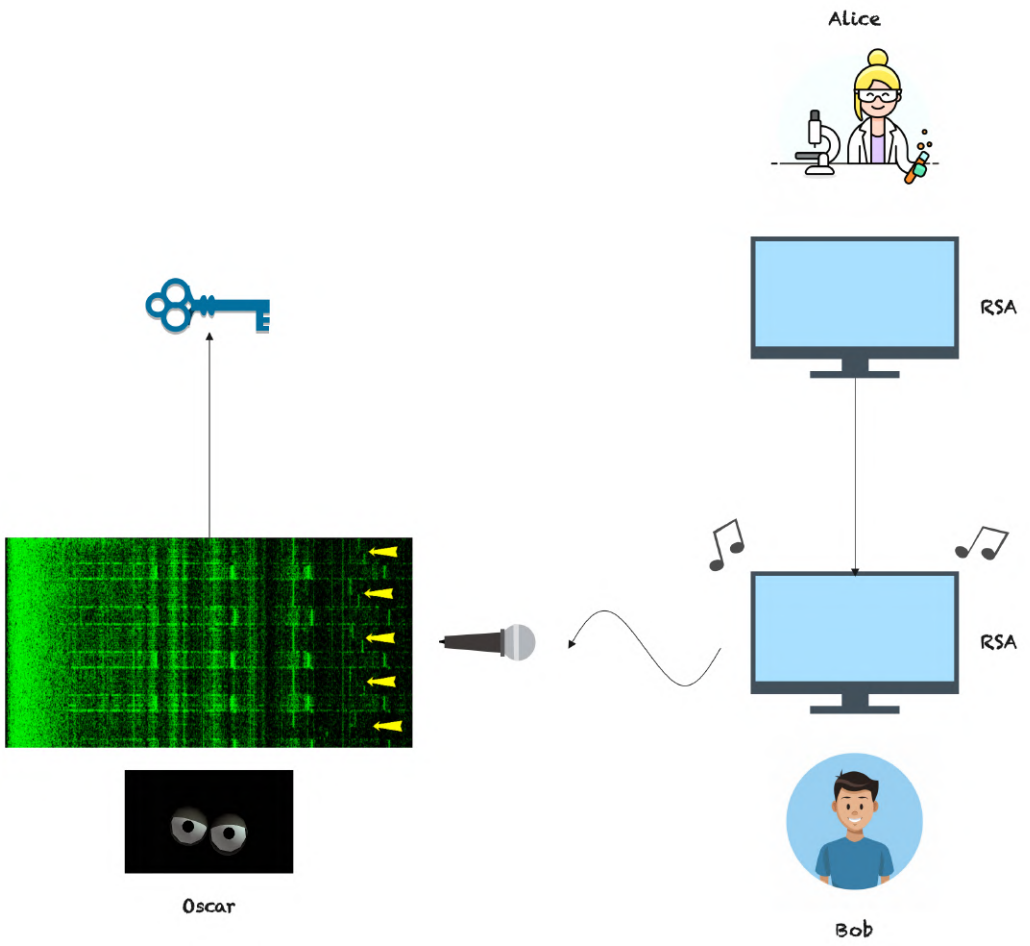


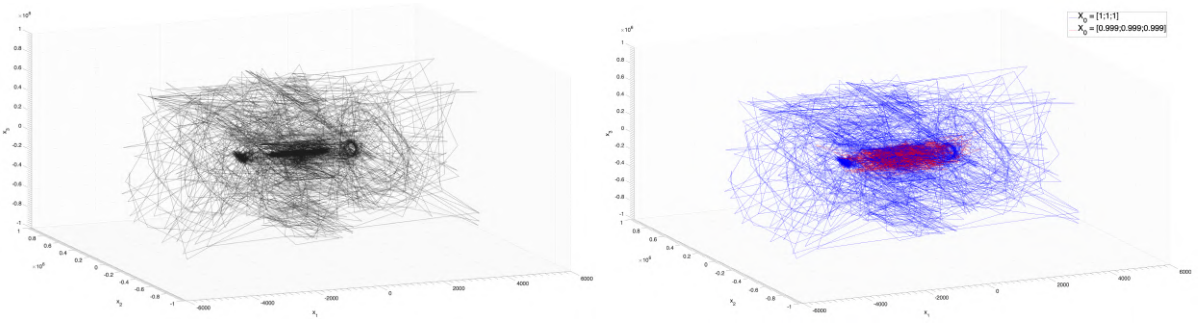
Figure 5.1: The acoustic cryptanalysis technique

5.2 The improved Lorenz system

The acoustic cryptanalysis method presented in [16] is shown in figure 5.1. Oscar uses the sound emitted by RSA decryption to find the secret key. The countermeasure I choose in this chapter is to use an improved Lorenz system in cascade with an ANFIS to mask the sound emitted by RSA.

The modified Lorenz system I use in this chapter is obtained using the method of adding noise to the state vector. Its state equation is given by 5.1.

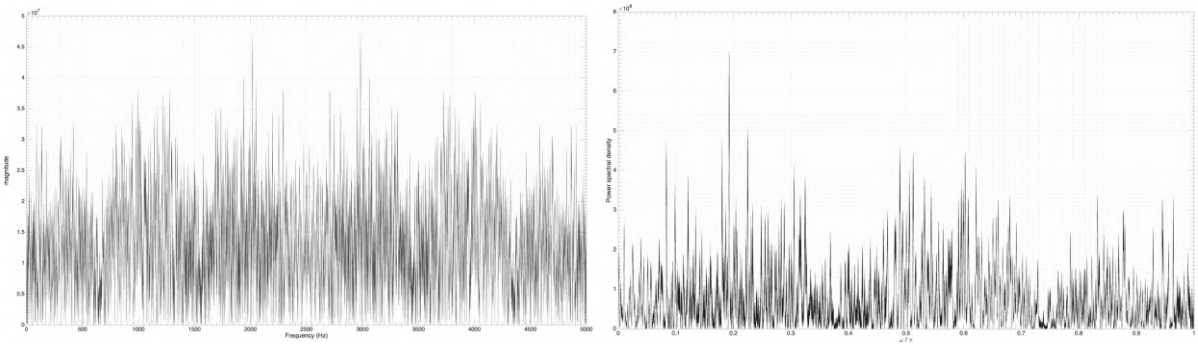
$$\begin{cases} \dot{x}_1 = 5\sigma(x_2 - x_1\frac{s}{10})\frac{s}{10} \\ \dot{x}_2 = x_1(\rho - x_3) - x_2\frac{s}{10} \\ \dot{x}_3 = x_1x_2 - 10\beta x_3\frac{s}{10} \end{cases} \tag{5.1}$$



(a) The improved Lorenz attractor

(b) The sensitivity to initial conditions

Figure 5.2: The improved Lorenz system



(a) The frequency spectrum of x_2

(b) The power spectral density of x_2

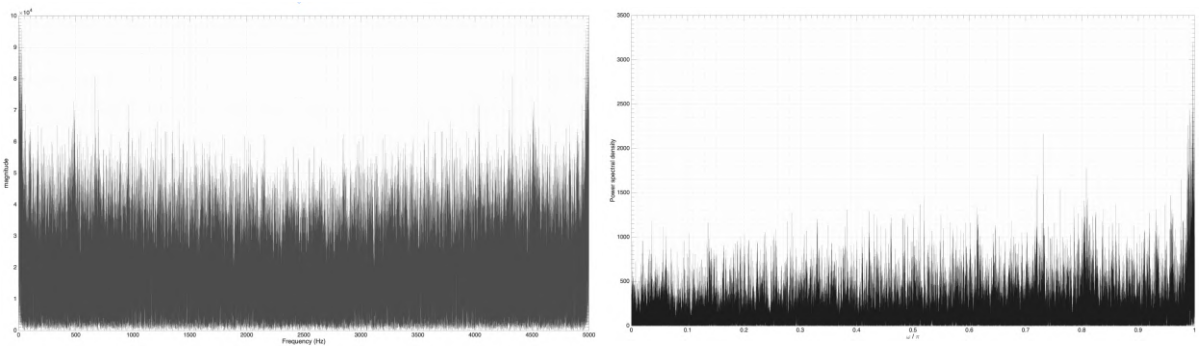
Figure 5.3: The characteristics of x_2

where s is obtained using the "random number" block of SIMULINK with standard parameters. I used the "auto" time step. The attractor of 5.1 is shown in figure 5.2a. As we can see in figure 5.2b, there is some sensitivity to initial conditions.

The signal that I want to use in this chapter is x_2 because it has good frequency characteristics, as it is shown in figures 5.3a and 5.3b.

5.3 The ANFIS training

The ANFIS that I use in this chapter has been trained to create a noise $\psi(x_2)$ given as an input the signal x_2. To train this ANFIS, I used as an input the signal x_2, and as an output, a white noise created using the SIMULINK block "band-limited white noise" with its standard characteristics. To obtain the data, I ran the simulation for $50s$ and I used the SIMULINK blocks "to workspace" with a sampling time of $0.005s$. I used the MATLAB command "anfisedit" to train the ANFIS. I used 3 triangular membership



(a) The frequency spectrum of $\psi(x_2)$

(b) The power spectral density of $\psi(x_2)$

Figure 5.4: The characteristics of $\psi(x_2)$

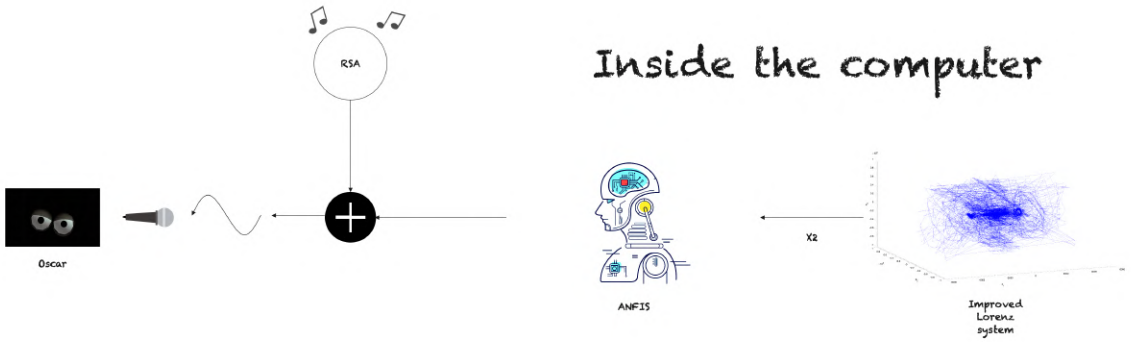


Figure 5.5: The ANFIS-Lorenz system

functions and I did 21 epochs. The results are shown in figure 5.4.

5.4 The ANFIS-Lorenz system and how to use it effectively

The way I combine the improved Lorenz system with ANFIS is shown in figure 5.5. The chaotic circuit generates the signal x_2 which is used as an input to the ANFIS. Then, the ANFIS constructs using x_2 a noise which is transformed into sound to mask the sound emitted by RSA.

One interesting question to ask is how to attack someone using the ANFIS-Lorenz system? If we construct the circuitry and the ANFIS, we will have the parameters of the ANFIS. So what we can do is to use x_2 as an input to a second ANFIS with the same parameters as the first one and then invert the signal to cancel it with the signal emitted

by the ANFIS which is inside the computer. The problem is that we need to have access to x_2. To achieve that, I thought of designing a port which emits an encrypted version of the electrical signal x_2. And then, we will use the right key to reconstruct the signal x_2. One way of encrypting x_2 is to use RSA, and because inside the computer there is the noise I created, this second RSA can not in theory be broken using acoustic cryptanalysis.

5.5 Conclusion

We have seen in this chapter a countermeasure against the acoustic cryptanalysis technique used against RSA and presented in [16]. I used an improved Lorenz system that I obtained using the method of adding noise to the state vector. I also used an ANFIS to make a chaotic synchronization attack difficult. I also presented a technique that allows the designer of the countermeasure to attack a user of the countermeasure. The next step consists of giving a practical realization of the circuits and programs and testing them in real conditions.

CHAPTER 6

CHAOTIC POST-QUANTUM LATTICE-BASED CRYPTOGRAPHY

6.1 Introduction

The objective of this chapter is to present two new potentially difficult problems on lattices. The first one is a variant of SVP (Shortest vector problem) to which I added an additional constraint that depends on the trajectory of a given chaotic system. The second problem is the SVP but on a chaotic attractor located inside a lattice. In either case, the parameters and initial conditions of the chaotic system used are kept secret, making the problem more difficult for Oscar as he will be missing an essential part of it. In other words, to solve these problems Oscar will have for the first case to solve an optimization problem without knowing one of the constraints, and for the second case, to solve an optimization problem on a space that he does not know.

6.2 The chaotic SVP (Shortest vector problem)

Given the Euclidean space $\mathbf{R}^n$, $n \in \mathbf{N}$. A lattice of $\mathbf{R}^n$ is any discrete finite subgroup of $\mathbf{R}^n$ of rank n. For a given lattice Λ of $\mathbf{R}^n$, we know that there exists a set of linearly independent vectors $b_i \in \mathbf{R}^n$ such that any point of Λ is a linear integer combination of the b_i's. This set is called a basis of the lattice. Conversely, a set of linear integer combinations of n linearly independent vectors of $\mathbf{R}^n$ is a lattice of $\mathbf{R}^n$. In other words, a

Figure 6.1: Lattices in art

lattice Λ of $\mathbf{R}^n$ is any set of vectors of $\mathbf{R}^n$ generated by the linear integer combinations of n linearly independent vectors $b_i \in \mathbf{R}^n$. More formally, it is defined by 6.1.

$$\Lambda = \{\sum_{i=1}^{n} \lambda_i b_i : \lambda_i \in \mathbf{Z}\} \tag{6.1}$$

The lattice structure is used in many areas such as art (figure 6.1) and crystallography (figure 6.2). We can see in figure 6.3 some crystals which are chemical elements having lattice structures.

In cryptography, lattices are used to generate problems considered to be mathematically difficult to solve. In asymmetric cryptography, the security of algorithms is based on the difficulty of solving integer factorization and discrete logarithm problems. With the advent of quantum computers, quantum algorithms for solving these problems have emerged such as Shor's algorithms [31]. The practical implementation of Shor's algorithms requires the use of thousands of qbits, which makes these algorithms unusable to date. But there is no guarantee that in a few years quantum technology will not have evolved to the point that we will be able to implement Shor's algorithms. It is why we need new cryptography methods that are robust to quantum attacks. Among these methods, we have the SVP (Shortest vector problem) which consists in finding, giving a basis (b_i) of a lattice Λ, the shortest non-zero vector in the sense of the Euclidean norm[1],

[1]It is also possible to use other norms.

Figure 6.2: Lattices in crystallography

i.e. the vector v satisfying 6.2.

$$L(v) = min\ ||w||_2 \tag{6.2}$$

where $L(v)$ denotes the length of v and the min is taken over all the vectors $w \in \Lambda^*$.

The SVP is known to be NP-hard. In my modification, I add a constraint that is dependent on the trajectory of a given chaotic system. This modification is shown in figure 6.4. The modified SVP is to find the shortest non-zero vector in the lattice that minimizes the difference between the chaotic trajectories if we take it as an initial condition to the chaotic system and we compare the result to the same chaotic system with an initial condition x_0. In other words, the problem is to find the shortest non-zero vector v in the lattice such that it minimizes $||\chi_{x_0} - \chi_v||$ where χ_{x_0} denotes the chaotic trajectory given the initial condition x_0.

6.3 The SVP on chaotic attractors inside lattices

The second problem I designed consists in the reduction of the space of the SVP optimization problem to a curve which is unknown to Oscar. This curve is a chaotic attractor. Obviously, the attractor is generally not a lattice. To prove it, it suffices to consider the

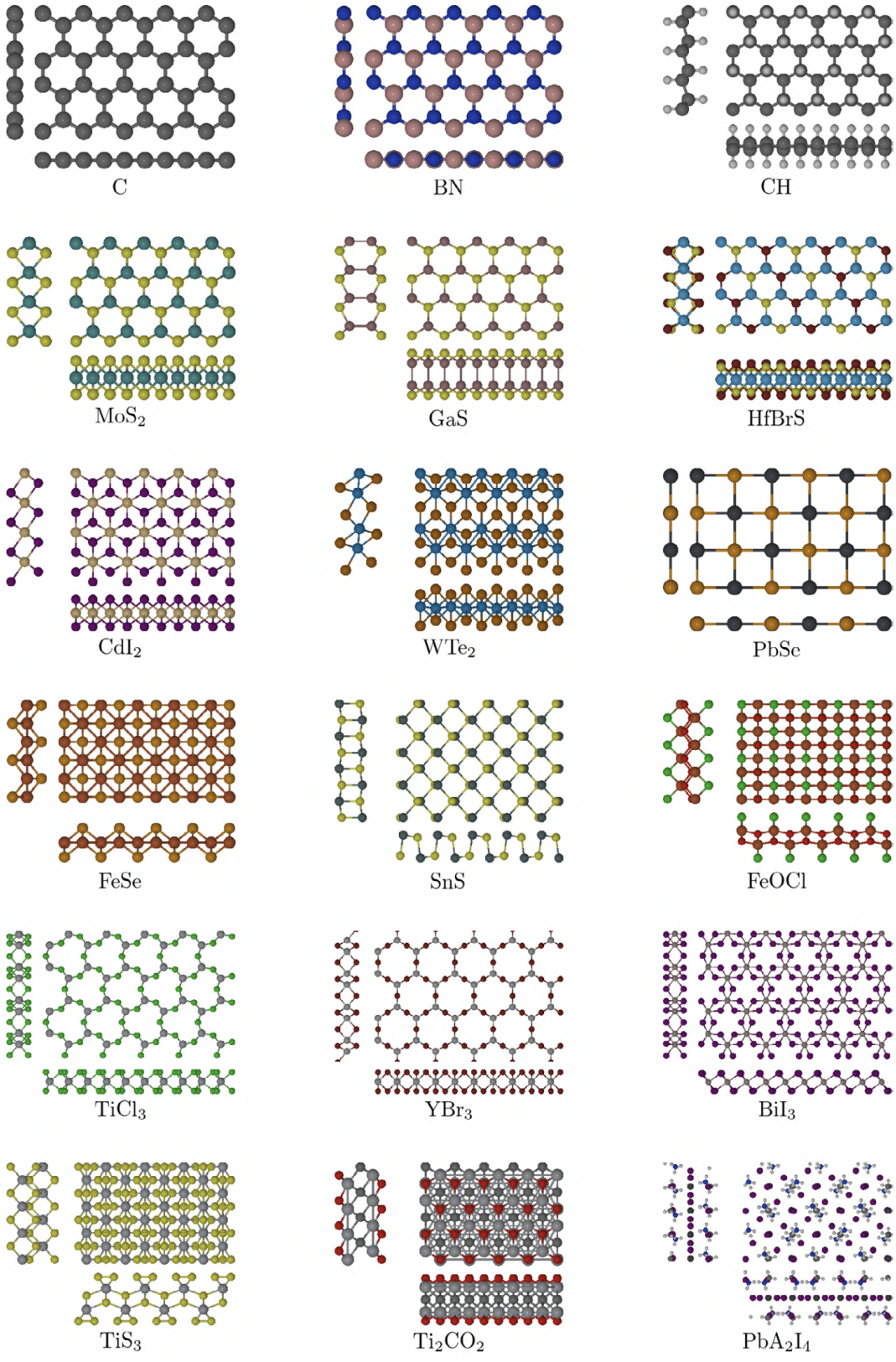


Figure 6.3: Some crystals

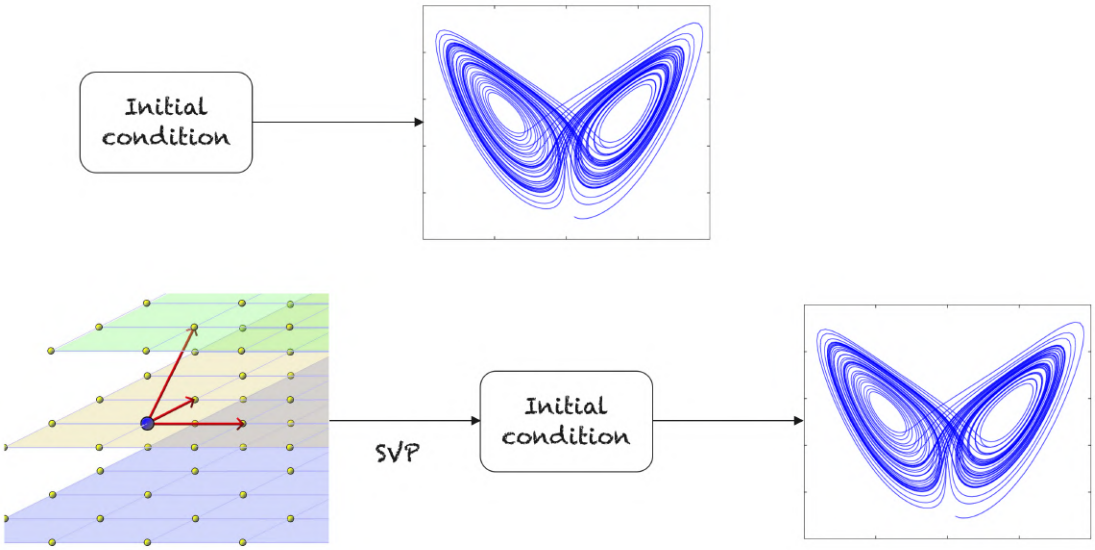


Figure 6.4: The chaotic SVP

first condition 6.3 to have an additive subgroup G.

$$a_1, a_2 \in G \Longrightarrow a_1 + a_2 \in G \tag{6.3}$$

This condition is generally not satisfied for vectors taken on an attractor (and on any other non-trivial curve of the space). The objective is therefore not to build a "chaotic" Lattice within an initial lattice. My goal is just to build a chaotic path within an initial lattice that one could take to resolve the SVP. More formally, the optimization problem now consists in finding the shortest non-zero vector of the lattice which lies on the chaotic attractor.

6.4 Conclusion

In this chapter, I have presented two new problems on lattices for uses in post-quantum cryptography. The first problem consists of the SVP with an additional constraint that takes into account the initial conditions and the trajectories of a given chaotic system. The second problem consists in reducing the space of the solutions of the SVP by considering only the solutions located on a given chaotic attractor located inside a lattice. If the parameters and the initial conditions of the chaotic system used are kept secret, then these two problems become theoretically unsolvable by Oscar since he does not know the essential part of them. The security of a cryptosystem using these problems would then in theory become unconditional, provided that the parameters of the chaotic system are kept

secret. Obviously, the problems presented in this chapter remain at the stage of ideas and thorough work must be done before being able to decide on the real effectiveness of the proposed problems.

CHAPTER 7

CONCLUSION

In this thesis, several points have been addressed. In chapter 3, I improved two observers built-in [12]. The first is an adaptive unknown inputs observer and the second is an adaptive sliding mode unknown inputs observer. The improvements I made induce a delay in the system. I built two cryptosystems that show that this delay is not necessarily problematic. I also developed a method to improve the frequency characteristics of chaotic systems.

In chapter 4, I improved a predictor-based super twisting second-order sliding mode observer presented in [18]. My improvement is to replace the discontinuous sign function with fuzzy inference systems. This approach has been used in a few articles such as [27, 28]. It eliminates the chattering effect which is due to the discontinuities of the sign function. I also introduced a technique that allowed the unknown input of a triangular system to be considered as a fictitious state, which allows an unknown input to be estimated using a state observer. This technique was illustrated by the use of the super twisting observer to estimate an unknown input. I also presented a modification of the higher-order sliding mode observers in which the order is now a function that varies according to an optimization criterion. I applied this adaptive order technique on the super twisting observer and I have shown through the simulations that the effect of the adaptive order was more and more present as we were advancing in the state vector.

In chapter 5, I presented a countermeasure to the acoustic cryptanalysis method against

RSA built-in [16]. My countermeasure is to mask the sound emitted by RSA decryption using a chaotic system with an ANFIS which is trained to emit white noise given a chaotic input. The role of ANFIS is to prevent Oscar from using chaotic synchronization techniques to eliminate masking.

In Chapter 6, I presented two modifications of the SVP on lattices. These changes consist of hiding an important part of the optimization problem from Oscar so that it is impossible for him to solve it. The first modification is to add a constraint to the SVP which depends on the trajectories of a chaotic system. The second modification consists of reducing the space of solutions of the SVP by making an intersection between the lattice and a given chaotic attractor. The initial parameters and conditions of the chaotic system are kept secret so that Oscar cannot solve these problems. The goal is to have unconditional security.

Here is now how I see the rest of this work. First of all, for the method of generating better chaotic systems, more simulations must be done to obtain attractors with good frequency properties, and possibly to detect a general structure on the optimal way of adding noise to the state vector. In addition, a more detailed mathematical analysis must be done. For example, it is necessary to calculate the Lyapunov exponents and the entropy of modified chaotic systems and compare the results obtained with the state of the art.

Secondly, a detailed construction of the cryptosystems presented in this thesis must be done and followed by a complete cryptanalysis to determine if the cryptosystems presented have a good level of security. Practical issues must also be taken into accounts, such as the circuit design of chaotic systems and the programming of observers.

We saw in chapter 4 that there were certain problems when using the Rossler system. We have to understand where they come from and how to solve them. A detailed mathematical study of the higher adaptive order sliding mode observers should also be made before they are used in practice. One idea would be to then use the higher adaptive order sliding mode observers for the synchronization of chaotic systems used in possible improvements of block ciphers such as AES.

Signal processing should be done on the signal obtained at the ANFIS output before

using it for chaotic masking. The amplitude is very variable and even if the frequency characteristics are good, it will be dangerous to directly add the sound emitted by RSA to the noise emitted by the ANFIS. The noise emitted by ANFIS must therefore be processed correctly before being used, and this processing must not deteriorate the frequency characteristics of the signal.

Finally, a mathematical analysis of the variations of the SVP proposed must be made. It is also necessary to seek which chaotic systems would be best suited to these problems.

BIBLIOGRAPHY

[1] G. ALVAREZ, L. HERNANDEZ, J. MUNOZ, F. MONTOYA, AND S. LI, *Security analysis of communication system based on the synchronization of different order chaotic systems*, Physics Letters A, 345 (2005), pp. 245–250.
Publisher: Elsevier.

[2] G. ALVAREZ AND S. LI, *Some basic cryptographic requirements for chaos-based cryptosystems*, International journal of bifurcation and chaos, 16 (2006), pp. 2129–2151.
Publisher: World Scientific.

[3] J.-P. BARBOT AND T. FLOQUET, *Iterative higher order sliding mode observer for nonlinear systems with unknown inputs*, Dynamics of Continuous, Discrete and Impulsive Systems, 17 (2010), pp. 1019–1033.

[4] G. BESANCON, *Remarks on nonlinear adaptive observer design*, Systems & control letters, 41 (2000), pp. 271–280.
Publisher: Elsevier.

[5] ——, *Nonlinear observers and applications*, vol. 363, Springer, 2007.

[6] J. CHEN AND H. ZHANG, *Robust detection of faulty actuators via unknown input observers*, International Journal of Systems Science, 22 (1991), pp. 1829–1839.
Publisher: Taylor & Francis.

[7] Y. M. CHO AND R. RAJAMANI, *A systematic approach to adaptive observer synthesis for nonlinear systems*, IEEE transactions on Automatic Control, 42 (1997), pp. 534–537.
Publisher: IEEE.

[8] K. M. CUOMO AND A. V. OPPENHEIM, *Circuit implementation of synchronized chaos with applications to communications*, Physical review letters, 71 (1993), p. 65.

Publisher: APS.

[9] M. DAROUACH, *Complements to full order observer design for linear systems with unknown inputs*, Applied Mathematics Letters, 22 (2009), pp. 1107–1111.
Publisher: Elsevier.

[10] M. DAROUACH, M. ZASADZINSKI, AND S. J. XU, *Full-order observers for linear systems with unknown inputs*, IEEE transactions on automatic control, 39 (1994), pp. 606–609.
Publisher: IEEE.

[11] O. DATCU, L. FRIDMAN, AND J.-P. BARBOT, *A third-order sliding-mode observer for a continuous delay chaotic system*, IFAC Proceedings Volumes, 45 (2012), pp. 175–180.
Publisher: Elsevier.

[12] H. DIMASSI, *Synchronisation des systemes chaotiques par observateurs et applications a la transmission d'informations*, PhD Thesis, Paris 11, 2012.

[13] C. EDWARDS, S. K. SPURGEON, AND R. J. PATTON, *Sliding mode observers for fault detection and isolation*, Automatica, 36 (2000), pp. 541–553.
Publisher: Elsevier.

[14] M. FARZA, M. M,ÄôSAAD, T. MAATOUG, AND M. KAMOUN, *Adaptive observers for nonlinearly parameterized class of nonlinear systems*, Automatica, 45 (2009), pp. 2292–2299.
Publisher: Elsevier.

[15] T. FLOQUET AND J.-P. BARBOT, *A canonical form for the design of unknown input sliding mode observers*, in Advances in variable structure and sliding mode control, Springer, 2006, pp. 271–292.

[16] D. GENKIN, A. SHAMIR, AND E. TROMER, *RSA key extraction via low-bandwidth acoustic cryptanalysis*, in Annual Cryptology Conference, Springer, 2014, pp. 444–461.

[17] K. S. HALLE, C. W. WU, M. ITOH, AND L. O. CHUA, *Spread spectrum communication through modulation of chaos*, International Journal of Bifurcation and Chaos, 3 (1993), pp. 469–477.
Publisher: World Scientific.

[18] A. HAMOUDI, N. DJEGHALI, AND M. BETTAYEB, *Predictor-based super-twisting sliding mode observer for synchronisation of nonlinear chaotic systems with delayed measurements*, International Journal of Systems Science, 51 (2020), pp. 3013–3029.
Publisher: Taylor & Francis.

[19] D. M. KATO AND M. EISENCRAFT, *On the power spectral density of chaotic signals generated by skew tent maps*, in 2007 International Symposium on Signals, Circuits and Systems, vol. 1, IEEE, 2007, pp. 1–4.

[20] S. KAWAJI AND K. SAWADA, *Observer design for linear descriptor systems with unknown inputs*, in Proceedings IECON'91: 1991 International Conference on Industrial Electronics, Control and Instrumentation, IEEE, 1991, pp. 2285–2288.

[21] L. J. KOCAREV, K. S. HALLE, K. ECKERT, L. O. CHUA, AND U. PARLITZ, *Experimental demonstration of secure communications via chaotic synchronization*, International Journal of Bifurcation and Chaos, 2 (1992), pp. 709–713.
Publisher: World Scientific.

[22] A. LEVANT, *Robust exact differentiation via sliding mode technique*, automatica, 34 (1998), pp. 379–384.
Publisher: Elsevier.

[23] ——, *Higher-order sliding modes, differentiation and output-feedback control*, International journal of Control, 76 (2003), pp. 924–941.
Publisher: Taylor & Francis.

[24] E. LORENZ, *Predictability: does the flap of a butterfly's wing in Brazil set off a tornado in Texas?*, na, 1972.

[25] C. PAAR AND J. PELZL, *Understanding cryptography: a textbook for students and practitioners*, Springer Science & Business Media, 2009.

[26] L. M. PECORA AND T. L. CARROLL, *Synchronization in chaotic systems*, Physical review letters, 64 (1990), p. 821.
Publisher: APS.

[27] C. PLATA, P. J. PRIETO, R. RAMIREZ-VILLALOBOS, AND L. N. CORIA, *Chaos Synchronization for Hyperchaotic Lorenz-Type System via Fuzzy-Based Sliding-Mode Observer*, Mathematical and Computational Applications, 25 (2020), p. 16.
Publisher: Multidisciplinary Digital Publishing Institute.

[28] P. J. PRIETO-ENTENZA, N. R. CAZAREZ-CASTRO, L. T. AGUILAR, S. L. CARDENAS-MACIEL, AND J. A. LOPEZ-RENTERIA, *A lyapunov analysis for mamdani type fuzzy-based sliding mode control*, IEEE Transactions on Fuzzy Systems, 28 (2019), pp. 1887–1895.
Publisher: IEEE.

[29] G. RINALDI, P. P. MENON, C. EDWARDS, AND A. FERRARA, *Higher order sliding mode observers in power grids with traditional and renewable sources*, IEEE Control Systems Letters, 4 (2019), pp. 223–228.
Publisher: IEEE.

[30] C. E. SHANNON, *Communication theory of secrecy systems*, The Bell system technical journal, 28 (1949), pp. 656–715.
Publisher: Nokia Bell Labs.

[31] P. W. SHOR, *Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer*, SIAM review, 41 (1999), pp. 303–332.
Publisher: SIAM.

[32] V. SYRMOS, *Observer design for descriptor systems with unmeasurable disturbances*, in [1992] Proceedings of the 31st IEEE Conference on Decision and Control, IEEE, 1992, pp. 981–982.

[33] B. WALCOTT AND S. ZAK, *State observation of nonlinear uncertain dynamical systems*, IEEE Transactions on automatic control, 32 (1987), pp. 166–170.
Publisher: IEEE.

[34] B. L. WALCOTT AND S. H. ZAK, *Combined observer-controller synthesis for uncertain dynamical systems with applications*, IEEE Transactions on systems, man, and cybernetics, 18 (1988), pp. 88–104.
Publisher: IEEE.

[35] S.-H. WANG, E. WANG, AND P. DORATO, *Observing the states of systems with unmeasurable disturbances*, IEEE transactions on Automatic Control, 20 (1975), pp. 716–717.

Publisher: IEEE.

[36] C. W. WU AND L. O. CHUA, *A simple way to synchronize chaotic systems with applications to secure communication systems*, International Journal of Bifurcation and Chaos, 3 (1993), pp. 1619–1627.
Publisher: World Scientific.

[37] A. XU AND Q. ZHANG, *Nonlinear system fault diagnosis based on adaptive estimation*, Automatica, 40 (2004), pp. 1181–1193.
Publisher: Elsevier.

[38] T. YANG, L.-B. YANG, AND C.-M. YANG, *Breaking chaotic secure communication using a spectrogram*, Physics Letters A, 247 (1998), pp. 105–111.
Publisher: Elsevier.

[39] Q. ZHANG, *Adaptive observer for MIMO linear time varying systems*, PhD Thesis, INRIA, 2001.

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

ECOLE NATIONALE POLYTECHNIQUE

Département d'Automatique

Laboratoire de Contrôle des Processus

Mémoire de projet de fin d'études
pour l'obtention du diplôme d'ingénieur d'état en Automatique

Synchronisation du chaos à base d'observateurs non-linéaires avec des applications en cryptographie

Auteur:
Mohamed Camil
Belhadjoudja

Encadrants:
Prof. Mohamed Tadjine
Dr. Messaoud Chakir

Jury

Président	Mr. M.S. BOUCHERIT	Professeur ENP
Examinateur	Mr. R. ILLOUL	MCA ENP
Encadrant	Mr. M. TADJINE	Professeur ENP
Encadrant	Mr. M. CHAKIR	Docteur ENP

ENP 2021