A MULTI MODULAR DYNAMICAL CRYPTOSYSTEM BASED ON CONTINUOUS-INTERVAL CELLULAR AUTOMATA

by

JESUS DAVID TERRAZAS GONZALEZ

A Thesis submitted to the Faculty of Graduate Studies of
The University of Manitoba
in partial fulfilment of the requirements of the degree of

MASTER OF SCIENCE

Department of Electrical and Computer Engineering
University of Manitoba
Winnipeg, Manitoba, Canada

THE UNIVERSITY OF MANITOBA

FACULTY OF GRADUATE STUDIES
****
COPYRIGHT PERMISSION

A Modular Dynamical Cryptosystem Based on Continuous Cellular Automata

BY

Jesus David Terrazas Gonzalez

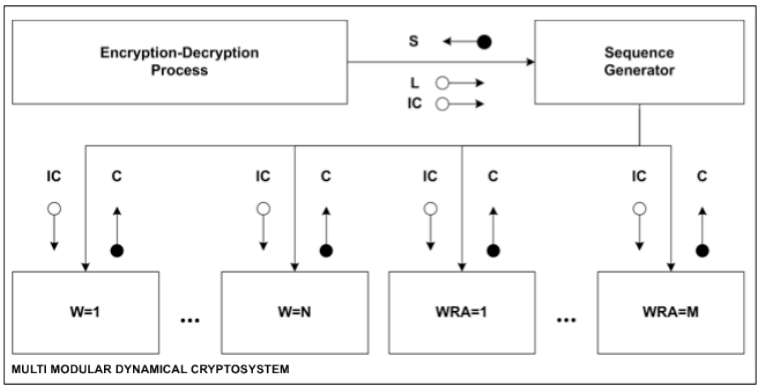
A Thesis/Practicum submitted to the Faculty of Graduate Studies of The University of Manitoba in partial fulfilment of the requirement of the degree

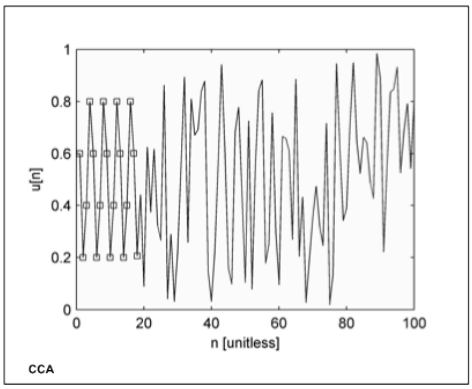
Of

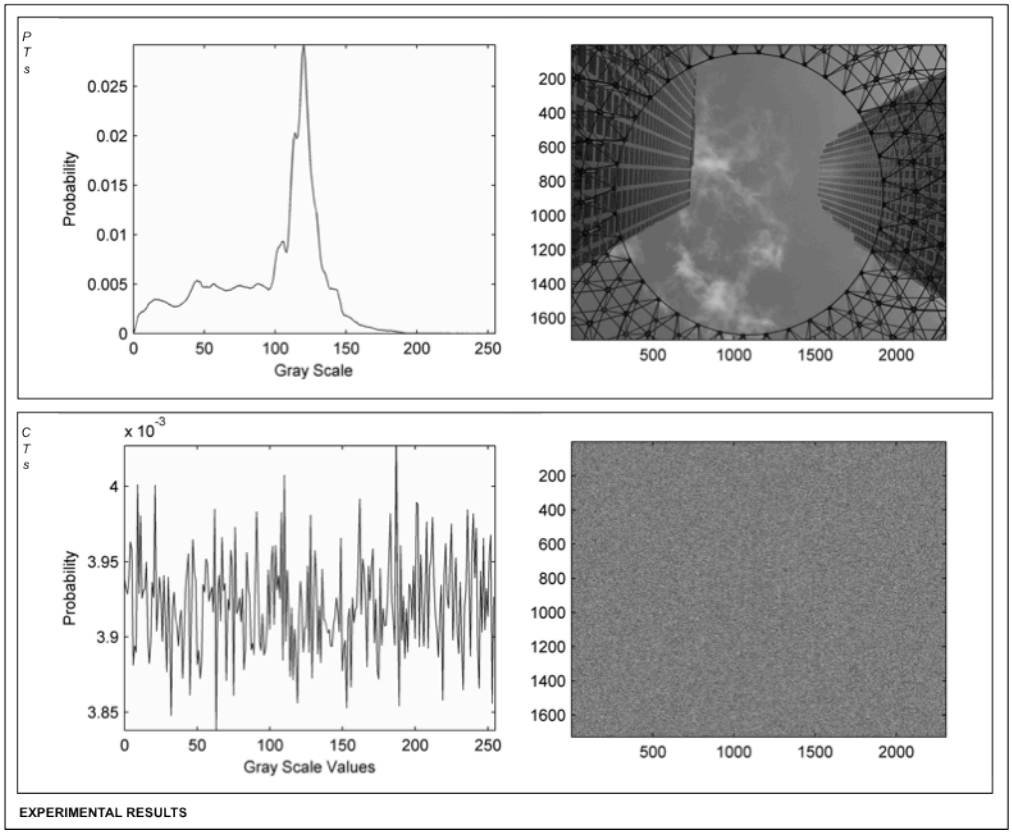
Master of Science

To Amparo

VISUAL ABSTRACT







ABSTRACT

This thesis presents a computationally efficient cryptosystem based on *chaotic continuous-interval cellular automata* (CCA). This cryptosystem increases data protection as demonstrated by its flexibility to encrypt/decrypt information from distinct sources (*e.g.*, text, sound, and images). This cryptosystem has the following enhancements over the previous chaos-based cryptosystems: (i) a mathematical model based on a new chaotic CCA strange attractor, (ii) integration of modules containing dynamical systems to generate complex sequences, (iii) generation of an unlimited number of keys due to the features of chaotic phenomena obtained through CCA, which is an improvement over previous symmetric cryptosystems, and (iv) a high-quality concealment of the cryptosystem strange attractor. Instead of using differential equations, a process of mixing chaotic sequences obtained from CCA is also introduced. As compared to other recent approaches, this mixing process provides a basis to achieve higher security by using a higher degree of complexity for the encryption/decryption processes. This cryptosystem is tested through the following three methods: (i) a stationarity test based on the invariance of the first ten statistical moments, (ii) a polyscale test based on the *variance fractal dimension trajectory* (VFDT) and the *spectral fractal dimension* (SFD), and (iii) a surrogate data test. This cryptosystem secures data from distinct sources, while leaving no patterns in the ciphertexts. This cryptosystem is robust in terms of resisting attacks that: (i) identify a chaotic system in the time domain, (ii) reconstruct the chaotic attractor by monitoring the system state variables, (iii) search the system synchronization parameters, (iv) statistical cryptanalysis, and (v) polyscale cryptanalysis.

ACKNOWLEDGEMENT

I would like to express my most profound and sincere thanks to those who provided me with the notion of an infinite number of infinities and the sense to tame chaos in engineering applications as well as personal life, especially my thesis advisor Prof. Witold Kinsner. He is a very unique person who greatly influenced my professional/research experience. His endless contributions to developing my research and interests have made a lasting impression in my life and they always will continue being an important pillar for me. My ideas, research procedures and protocols, study approaches, problem-solving techniques, have been strongly modeled under his guidance. I have enjoyed being his student and feel honoured to have had the opportunity to collaborate with him in different projects.

I am also thankful for the contributions of the past and present members who have and constitute the Delta group. This group has cultivated one of the friendliest environments where open knowledge sharing is the recognized key to grow as a researcher. I encourage the future members of this group and researchers in general to pursue their deepest passion in their research. This is one of the many ways that opens the road to success and makes it a gratifying experience. Never do things because you are expected to do them, do them because you are deeply passionate in your hearth about them. I have that feeling with the work performed across this thesis.

I bring my deepest respect to my parents, especially to my beloved mother Amparo González García, whom since I was a kid stamped in my mind the idea of being a good man during my life on this planet. Her delicate preparation of my things to attend school for the first time ever is something I will always remember and be thankful for because my inspiration and

encouragement source to look for success always is her. I also recognize the effort that my father Gerónimo Terrazas López has made in order to see me obtain my personal goals. My parents have been very supportive when I left Mexico to pursue my dreams abroad. They taught me that life success is not present in material things, but in the positive impact we can have through our actions.

I thank all the members of my family for sharing life-lasting experiences with me. I credit my sisters for their encouragement, and believing in my capabilities. Their loyal words and emotional support always have been given me great strength in many situations across my life.

My supportive friends that were there when I needed a helping hand and supportive words also deserve my gratitude.

Lastly, but most importantly, I wish to recognize God for providing the invisible hand that always has been my support. The most remarkable doctrine that I have obtained across my life is that God has been, is, and will be always capable of transforming my adversities into victories. I also thank God for the gift of infinities and allowing me to play with some of them through chaos phenomena.

Great is our Lord, and of great power: His understanding is infinite.

Psalms 147:5

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AES	**A**dvanced **E**ncryption **S**tandard
ASCII	**A**merican **S**tandard **C**ode for **I**nformation **I**nterchange
BB84	**B**ennett and **B**rassard protocol published in 19**84**
CA	**C**ellular **A**utomata
CCA	**C**ontinuous **C**ellular **A**utomata
CIA	**C**onfidentiality, **I**ntegrity, and **A**vailability (used in computer security)
CT	**C**ipher**t**ext
DCT	**D**iscrete **C**osine **T**ransform
DE	**D**ifferential **E**quation
DES	**D**ata **E**ncryption **S**tandard
DPA	**D**ifferential **P**ower **A**nalysis
ECC	**E**lliptic-**C**urve **C**ryptography
ECDLP	**E**lliptic-**C**urve **D**iscrete **L**ogarithm **P**roblem
ECDSA	Analogue of the US government digital signature standard
ElGamal	A cryptographic algorithm in ECC
FBI	**F**ederal **B**ureau of **I**nvestigations
FDMA	**F**requency **D**ivision **M**ultiple **A**ccess
FFT	**F**ast **F**ourier **T**ransform
FSM	**F**inite **S**tate **M**achine
FSS7	**F**inite **S**ense **S**tationarity **7**
FSS10	**F**inite **S**ense **S**tationarity **10**

GCHQ	**U**nited **K**ingdom **G**overnment **C**ommunications **H**eadquarters
GF	**G**alois Finite **F**ield
GWN	**G**aussian **W**hite **N**oise
ICS	**I**ndustrial **C**ontrol **S**ystem
IT	**I**nformation **T**echnology
I/O	**I**nput/**O**utput
LFSR	**L**inear **F**eedback **S**hift **R**egister
Matlab	**Mat**rix **Lab**oratory
mpmf	**M**arginal **p**robability **m**ass **f**unction
Mathematica	Simulation software developed by Wolfram Research
MIM	**M**an **i**n the **M**iddle
MMDC	**M**ulti-**M**odular **D**ynamical **C**ryptosystem
MQV	**M**enezes-**Q**u-**V**anstone protocol
NBS	**N**ational **B**ureau of **S**tandards
NCS	**N**etworked **C**ontrol **S**ystems
NIST	**N**ational **I**nstitute of **St**andards
NSA	**N**ational **S**ecurity **A**gency
PCA	**P**rogrammable **C**ellular **A**utomata
PDE	**P**artial **D**ifferential **E**quation
pdf	**p**robability **d**istribution **f**unction
PKC	**P**ublic-**K**ey **C**ryptography
PLC	**P**rogrammable **L**ogic **C**ontroller

pmf	**P**robability **m**ass **f**unction
PRNG	**P**seudo-**R**andom **N**umber **G**enerator
PSD	**P**ower **S**pectrum **D**ensity
PT	**P**lain**t**ext
q-bit	**Q**uantum **Bit**
QC	**Q**uantum **C**ryptography
RBN	**R**adio **B**ackground **N**oise
RGB	**R**ed, **G**reen, and **B**lue channels used in images
RFID	**R**adio **F**requency **Id**entification
RSA	**R**ivest-**S**hamir-**A**delman PKC algorithm
SFD	**S**pectral **F**ractal **D**imension
S	Encryption/decryption **s**equence generated by the MMDC
SG	**S**equence **G**enerator
SS-CDMA	**S**pread **S**pectrum **C**ode **D**ivision **M**ultiple **A**ccess
SSS	**S**trong **S**ense **S**tationarity
TMDA	**T**ime **D**ivision **M**ultiple **A**ccess
TRNG	**T**rue **R**andom **N**umber **G**enerator
var	**Var**iance operator
VFD	**V**ariance **F**ractal **D**imension
VFDT	**V**ariance **F**ractal **D**imension **T**rajectory
VLSI	**V**ery **L**arge **S**cale **I**ntegration
W	**W**orkers

WRA **W**orkers **R**andomly **A**ctivated

WSS **W**eak **S**ense **S**tationarity

NOTE

The spelling used in this thesis follows the Canadian rules, and not the American or British.

LIST OF SYMBOLS

$\mid$	OR operator
$\oplus$	Exclusive OR operator
$\lfloor \bullet \rfloor$	Floor operator
$\in$	Element of a set
$\sim$	Proportional to
$\uparrow$	q-bit that represents 0 in the V-H orthogonal basis
$\rightarrow$	q-bit that represents 1 in the V-H orthogonal basis
$\nearrow$	q-bit that represents 0 in the D-C orthogonal basis
$\nwarrow$	q-bit that represents 1 in the D-C orthogonal basis
$\lvert + \rangle$	q-bit that represents 0 in classical computing systems
$\lvert - \rangle$	q-bit that represents 1 in classical computing systems
$\lvert 0 \rangle$	q-bit that represents 0 in classical computing systems
$\lvert 1 \rangle$	q-bit that represents 1 in classical computing systems
α	Integer in the multiplicative group of the integers modulo φ
β	Spectral exponent, f^{β}
γ	Parameter computed iteratively to find α in the ElGamal algorithm
δ	Decryption exponent in the RSA algorithm
δt	Time interval between two consecutive samples (finite)
ΔB	Amplitude increment (finite)

Δt Time increment (finite)

Δx Line run $\left(x_2 - x_1\right)$ to determine its slope

Δy Line rise $\left(y_2 - y_1\right)$ to determine its slope

ε Encryption exponent in the RSA algorithm

η Modulus obtained through the product of two large and distinct primes

κ *Bit*-length to find φ in the ElGamal algorithm

υ One of the two large primes in RSA used to generate the modulus η

ϖ Order of the cyclic group G

π Pi (3.14159265…)

ρ One of the two large primes in RSA used to generate the modulus η

σ Standard deviation

σ_{as} Standard deviation after smoothing

σ_{os} Standard deviation before smoothing

σ^2 Variance

ς Private key in ElGamal (random integer)

τ Index with the value of the current window length n_w

φ Large random prime used as a component of the public key in ElGamal

χ^2 Chi squared statistical test

ψ Security parameter used in ElGamal

a Element of the set of CCA used in the cryptosystem modules

$\mathbf{A}^H\mathbf{A}$ Grammian operator

b Logarithm base

c Element of the ciphertexts set

$\mathbf{c}$ Coefficients to minimize the weighted square error

$B(t)$ Continuous signal under analysis to determine the VFD or VFDT

C Contribution sequence

d Element of the set of decryption rules

$d_K[\bullet]$ Decryption rule

$d_{K_{RSA16}}[\bullet]$ Decryption rule for the RSA16

D_β Spectral fractal dimension

D_σ Variance fractal dimension

D – C Diagonal-counter diagonal orthogonal basis

e Encryption rules space

E Euclidean dimension

$e_K[\bullet]$ Encryption rule

$e_{K_{RSA16}}[\bullet]$ Encryption rule for the RSA16

$e_{K_{RBN}}[\bullet]$ Encryption rule for a cryptosystem based on a RBN

$f[\bullet]$ Boolean function

f^β A given frequency to the power of the spectral exponent β used in SFD

f_{SL} Significance level

g Real parameter for the multi needle attractor

G	Cyclic group
$\mathrm{GF}(q)$	Galois finite field of q elements
H	Hurst exponent
int	Integer part of a decimal number
j	Index used to create l non-overlapping windows
k_π	Number of cycles in the main loop to calculate the VFD
key	Piano key
key_a	Piano key reference
k	Element of the set of possible keys in the cryptosystem
K_{buf}	Number of points in the log-log plot that are discarded to obtain the VFDT
$K_{\max}$	Maximum number of points that is possible to include in the log-log plot to obtain the VFDT
K_{hi}	Maximum number of points in the log-log plot that are considered to obtain the VFDT
K_{low}	Minimum number of points in the log-log plot that are considered to obtain the VFDT
l	Number of non-overlapping windows
L	Length of a sequence useful for encryption
lim	Function limit
$\log_b$	Logarithm base b
m	Line slope

M Number of points in a rectangular averaging filter

n Sample index in a time series

n_k Number of samples used to calculate the VFD at a given cycle k

n_w Window length

N_k Number of windows in the signal where a VFDT is being calculated

N_T Sample space

o Sample number. Auxiliary index for *two*-dimensional CA

O Point at infinity

p Element of the set of plaintexts

$p[\bullet]$ Original information sequence

P_C Tones pitch contained in the piano C scale

$P(f,T)$ Power spectrum density

$P(key_a)$ Reference pitch in Hertz

$P(key)$ Pitch of a piano key in Hertz

r Dynamic range

s Slope

S Sequence chaotically mixed

$\mathbf{S}_n$ Set of vectors in a phase reconstruction

$trng[\bullet]$ Generator of true random numbers

$\mathcal{T}$ Sixtuple that defines the multi-modular dynamical cryptosystem based on CCA

$\mathcal{T}_S$ Quintuple that defines a cryptosystem according to Stinson

T Total time over which a sample space is obtained

$u[\bullet]$ Discrete map obtained using a CCA

V – H Vertical-horizontal orthogonal basis

$\mathbf{W}$ Weighting matrix

$x[n]$ Input signal of a rectangular averaging filter

X_k Horizontal value in the log-log plot at a given cycle k

X_t A given time series

(x,y) Cartesian coordinates for horizontal x (*abscissa*) and vertical y (*ordinate*) components

$y[n]$ Output signal of a rectangular averaging filter

$\mathbf{y}$ Vector containing the values for each averaged interval of frequencies

$\hat{\mathbf{y}}$ Projection vector

Y_k Vertical value in the log-log plot at a given cycle k

$\mathbb{A}$ Set of CCA

$\mathbb{C}$ Set of ciphertexts

$\mathbb{D}$ Set of decryption rules

$\mathbb{E}$ Set of encryption rules

$\mathbb{K}$ Set of possible keys in the cryptosystem

$\mathbb{N}$ Set of natural numbers

$\mathbb{P}$ Set of plaintexts

$\mathbb{R}$ Set of real numbers

$\mathbb{Z}_{\varphi}^{*}$ Multiplicative group of integers modulo φ

CHAPTER I

INTRODUCTION

Everything should be made as simple as possible but not simpler.

— Albert Einstein

1.1 Motivation and Problem Definition

Cryptosystems have been developed to handle the challenging task of data protection in the modern information era. The purpose of cryptography is to hide the contents of messages to make them unrecognizable, except to someone who has the decryption method available [AnIS08] and more importantly, the key. Cryptosystems enhance the protection of data against low (*e.g.*, buffer overflow) and high level (*e.g.*, SQL injection) attacks. Different cryptosystems have been proposed and implemented in either hardware [AnSI07], [AnIS08], or software [ASRI08], or mixtures of both. Cryptosystems based on *cellular automata* (CA) [ASRI08] are preferred over continuous chaotic systems (*e.g.*, [MoSb10] and [YiRY09]) because of the simplicity and speed of CA-based computations, in contrast to the more costly equivalent models based on *differential equations* (DE). Traditional cryptosystems can be expensive in terms of configuration, storage space, bandwidth, and processing time. Also advanced cryptosystems, as is the case of the ones based on quantum cryptography, require expensive and highly specialized

hardware in order to function ([Assc06] and [MePS11]). Systems similar to CA were studied in the late 1950s to generate random sequences in cryptography [Wolf02], but they have not been fully explored.

The cryptosystem proposed is based on *continuous-interval cellular automata* (CCA) that are generalized CA. The specific interval considered is $[0,1]$. It is shown that this cryptosystem is very fast and highly secure when compared with alternative cryptosystems (*e.g.*, RSA, ElGamal, and a cryptosystem based on radio background noise). Also, it is applicable to distinct classes of data, including text, sound, and images.

The degree of complexity of a dynamical system or a cryptosystem based on CA has not been measured in the past in the reported literature. This thesis presents such a complexity measure based on the *variance fractal dimension trajectory* (VFDT) and *spectral fractal dimension* (SFD) ([Kins07c] and [KiGr08]). Both methods measure the complexity of the CCA chaotic sequences and compare them with the characteristics of white noise. A method to precisely determine the window size in which a signal can be considered strongly stationary is also presented. This method provides a visualization tool to determine this value in a compact form. In the same way, different flaws present in cryptosystems based on chaos phenomena like [YaWC97]: (i) reconstructing the chaotic attractor by means of finding state variables, (ii) attempting to build a model of the system, and (iii) exploiting synchronization patterns are eliminated through the use of surrogate data. The cryptosystem proposed in this thesis is an endeavour to obtain highly secure and at the same time compact computational technologies to protect data against attackers.

Cryptography is successful if the encoded information cannot be broken, if it is

computationally efficient to encrypt the data [Stin06], and at the same time if it is computationally very costly or extremely difficult to decrypt it. Security is an important, challenging, and multi-dimensional research field in networked computing and communication systems [AlBa11]. Cryptography is one of the many aspects of network security, including: access control, security protocols, information and hardware security, privacy, risk management, and resource allocation, among the most important ones [AlBa11]. It is a good practice to identify potential non-secure points, new exploits and tools, and the correct time to implement changes [PaBe09].

Given the dynamic nature of network security [AlBa11] one should not rely on static measures, or computationally costly algorithms. Dynamical problems require dynamical solutions [Kins07b]. Among the approaches considered in this thesis to design dynamical cryptosystems one provided by Shannon states that: *"Good mixing transformations are often formed by repeated products of two simple non-commuting operations. Hopf has shown, for example, that pastry dough can be mixed by such a sequence of operations. The dough is first rolled out into a thin slab, then folded over, then rolled, and then folded again, etc."* [Shan49]. This idea is the core in the interaction among modules containing a CA to perform the chaotic mixing of different complex behaviours where simple operations are of paramount importance. Implementations based either on high-speed hardware, or software, or a hybrid of these are easier to deploy and are more appealing to the industries developing products with limited computational capabilities (*i.e.*, smartcards and smartphones) requiring data protection for their large number of users.

Past computing and communication systems have not been designed with security as a

priority (*e.g.*, [Zhao10] and [SuHC10]). Computational systems available today are based on costly data protection algorithms that are vulnerable to different cryptanalysis attacks like: brute force that relies on the computing power that is available to the attacker; timing attacks against *public-key cryptography* (PKC) cryptosystems [MMTS07], as is the case of RSA implementations in smartcards; *differential power analysis* (DPA) [KJJR11] exploiting integrated circuits (IC) based on power consumption; side-channel analysis and fault injection are a threat against embedded cryptographic devices including *radio frequency identifiers* (RFIDs) [KaOP2010]; electromagnetic (EM) analysis that measures the EM field radiation of small regions in ICs ([Tana07] and [HMHS12]) to correlate them with the data that they are processing. This cryptanalysis technique has been proven successful in attacks against *elliptic curve cryptography* (ECC) implementations [HMHS12], to mention one of a few. Since it is practically impossible for a security expert to oversee all systems all the time [AlBa11], the development of more robust protection tools is required. Data from sensors or commands sent through insecure channels to actuators require protection by properly identifying and authenticating the source (*i.e.*, subject or object), which is requesting the execution of a task in highly secure applications.

Industrial control systems (ICS) are used world wide in manufacturing, assembly lines, refineries, power generation plants, water management, oil and gas pipelines, wind farms, airports, space stations, and buildings to mention a few [Karn11]. Most of them use *supervisory control and data acquisition* (SCADA) [Lang11] systems schemes that rely on *programmable logic controllers* (PLC), which are the specialized computers controlling automated physical processes.

In a nutshell, a SCADA system is an application that allows human operators to monitor an industrial process and to store and analyze process values [Lang11]. Programmable logic controllers have input/output (I/O) arrangements for various applications in physical environments. They normally have sensors on the inputs and the outputs typically operate equipment such as motors, switches, and relays [ChAb11]. Also, PLC's could have specialized controllers attached. Specialized controllers are usually small real-time computer systems manipulating electrical outputs based on the condition of electrical input signals and program logic. Devices such as pumps, valves, motor drives, thermometers, and tachometers are electrically connected (hardwired) to a controller, either directly of by what's called a *fieldbus connection*. While a computer program only operates on information, a controller program, also known as *ladder logic*, operates interfaced directly on physics (*e.g.*, control of AC motors). Manipulations of a controller traditionally have less to do with the information *confidentiality*, *integrity*, and *availability* (CIA) [PfPf11] and more to do with the performance and output of a physical production process in a given industry [Lang11].

Most commonly, in many cases ICSs are programmed and maintained by electricians rather than *information technology* (IT) personnel. Often engineers that are related with ICS have any experience in IT security. The average PLC does not count with: identification/authentication/authorization processes, antivirus software, hard disk, or patch management. Vulnerabilities in traditional ICSs could be legitimate product features that cannot be patched in a brief time [Mcgr11]. It is because of these reasons that different industries worldwide have started to get concerned about data security in their processes. This demand is expected to grow. A remarkable phrase, created by Constantine [Cons11] in the computer

security research community, which should not be ignored is: *"Anything that can be turned on under program control can be turned off; anything that can be controlled remotely can be thrown out of control remotely."* This fact becomes of paramount importance to maintain vital facilities around the globe operating in secure conditions because achieving this data security is transformed into security for human lives. A very well documented case in which an ICS is targeted has been presented by Matrosov *et al.* [MRHM12].

Secure communications for sensitive information is not only compelling for military and government institutions, but also for non-military industries, businesses, and individual needs [AnIS08]. Today, we can obtain many e-Services from sources such as e-Business [TaZh05], e-Commerce [QuMZ08], e-Health ([DiKl10] and [BoBC09]), e-Government [MoSb10], and e-Goods [NZSC05]. Such e-Services demand a high degree of availability and information confidence, thus requiring robust methods capable of offering high reliability in data security.

The creation of Internet connected networks that span continents and oceans; reach into land, air, and space vehicles; and confront microcomputers as well as large mainframe computers [HiHS83]. There are many different protocols critical to the communication of data through the Internet. These infrastructure protocols were developed early in the history of the Internet, when the risk environment was much more benign and the threat environment was not yet fully understood, and they are not fully protected against attacks to disclose confidential information. Cryptography, the art of secret communication, is an indispensable part in Internet communications [YiRY09] (*e.g.*, cloud computing services [HuKl09]) demanding technological advances to protect data in a secure manner. In fact, cryptography provides locks of different degrees of strength and keys to isolated or collaborative networks in the information age

[Sing99]. Services based on cloud computing rely highly in the most advanced encryption techniques to ensure data safety [HuKl09]. Every type of digital protection that we can think of relies on cryptographic techniques at some point to provide services of authentication, privacy, integrity, and confidentiality. In addition, commands protected by means of encryption initially used in the military fields are now migrating to different civilian industries.

The broadly used web services bring about many new security problems. Some approaches to manage the access control rely on poor ways to enforce authentication like feedback [JiPe10] from honest and unauthorized accesses. This implies that a computational system might not be capable of blocking undesired accesses becoming then vulnerable. In fact, a computational system could be threatened by attackers just because of its existence. The need of web-based systems that reduce unauthorized accesses is of vital importance for their users. Encryption is essential to protect our privacy and guarantee the success of the digital marketplace [Sing99].

A telerobotic system is controlled by means of a feedback control system where the control loop is closed via some form of communication networks [ZoGu10]. This is known as *networked control system* (NCS), which integrates distributed sensors, controllers, actuators, networks, and ICS. The networked control system is the infrastructure that allows the transmission of supermedia [EXKL03], which is the collection of commands and sensory feedback streams, between the different entities of the telerobotic system. Robotics researchers have been using the Internet and web services as a means to provide feedback for teleoperations and user interfaces. Internet-based teleoperations will inevitably lead to many useful applications in various sectors of our modern society. Surgeons may one day remotely operate on patients despite the distance between them. However, computer security in *networked control systems* (NCS) is not

universally known by researchers and consequently many developments in telerobotics have poor security layers. Past design of control systems almost did not have any application of security technology. Most of them applied simple techniques to prevent an unauthorized access, such as very limited codes and keys. The common NCS is always composed of equipment produced by different manufacturers and the integrators always do not consider security design for the whole system. In addition, the users of the NCS are not security experts and they do not know how to make the system more secure. Being introduced into control systems, the Internet brings much convenience on one hand, but on the other it makes the system more vulnerable to attacks [YCLS09]. Industrial spies can remotely access confidential information or production commands of the key instruments and equipment; malicious hackers intercept, tamper, forge, and retransmit data information and production commands transmitted over networks [ZoGu10]. Applications in Internet based teleoperations definitively require the most advanced methods available in data protection. These methods have to be inherently highly secure and computationally inexpensive. The Internet has started a transition from the *Internet Protocol version 4* (IPv4) to IPv6 [ATHB11] in June 8, 2011. However, once this transition is finished IPv6 is not a guarantee for freedom against threats.

1.2 Cellular Automata

Chaotic phenomena can be found in nature (*e.g.*, the weather and the Jupiter red spot [Stro00]) and modeled through computationally costly differential equations (*e.g.*, Lorenz attractor [Lore63]). Cellular automata are capable of developing chaotic behaviour using simple operations or rules [Wolf02], thus offering the benefit of high-speed computation. This is the basis in the development of the *multi-modular dynamical cryptosystem* (MMDC). “Modular” is

used in the sense that different modules, each containing a CCA-based dynamical system, are capable of interacting to generate a complex sequence. It is important that a cryptosystem has the flexibility to encrypt different datasets, rather to focus on specific ones [MoSb10]. Fast computation is also desired when securing data. This wanted feature applies either for key generation or encryption/decryption processes. This research provides a substantial contribution to high-speed implementation and data security of cryptosystems.

1.3 Research Questions

The importance of cryptosystems for our digital society is undeniable. In fact, this modern society depends on cryptography and it could not maintain its current operations without the protocols based on cryptosystems. Cryptosystems allow users to have ideally a high degree of security while conducting financial operations through the Internet. In the same way, military agencies giving instructions cover their operations by using encrypted communications systems. However, the different cryptosystems derived from PKC and ECC require a high amount of computational power to function. Holding these facts into account, the question is, can an incredibly simple system that is based on CCA support complex behaviour to construct a useful cryptosystem capable of protecting data? Once a cryptosystem based on CA is developed, is this cryptosystem capable of protecting different types of data in a similar degree?

Cryptosystems require a stationary, statistically invariant, source of randomness. Conventionally, a source of randomness that is stationary in the *weak sense stationarity* (WSS) is acceptable. Can the cryptosystem developed here provide the necessary base or even surpass this requirement by being invariant in the first ten statistical moments or *Finite Sense Stationarity 10* (FSS10)? Which is the correct method to determine the stationarity window for the

cryptosystem?

Chaotic systems based on DEs (*e.g.*, [MoSb10] and [YiRY09]) have been widely used in cryptosystems design. The set of nonlinear equations used in these chaotic systems (*i.e.*, [Lore63] and [YaWC97]) make them computationally costly because of the overload of computer systems resources. Could an incredibly simple system based on CA exhibit similar behaviour, chaos-like, than chaotic systems based on DEs?

Cryptosystems previously designed based on chaos theory could exhibit known vulnerabilities like identification of a chaotic system in the time domain, model reconstruction by monitoring state variables, and synchronization search. Could a cryptosystem based on CA overcome these vulnerabilities that traditional chaotic cryptosystems cannot? Furthermore, is it possible to provide stealth capabilities to the chaotic attractor of a cryptosystem based on CA to make the cryptosystem robust for cryptanalysis?

The formal thesis question for the work presented is

Can a simple cryptosystem based on CA exhibit chaotic behaviour that is FSS10 stationary, have no manifestation of classical vulnerabilities of chaos based cryptosystems and, resisting strongly cryptanalysis attacks?

1.4 Statement of Objectives of the Thesis

The prime goal in this thesis is to provide a cryptosystem capable of being fast to encrypt/decrypt, highly secure, and capable of protecting data from different sources. While all these are provided, the cryptosystem is demonstrated to be inherently FSS10 stationary and capable of eliminating known vulnerabilities of chaotic cryptosystems.

The requirements considered in order to achieve this prime goal are:

a) Generate a simple mathematical model using chaos theory that can provide infinite sequences useful for encryption. This is accomplished and defined by the equation presented in (2.2).

b) Integrate the mathematical model obtained in multiple modules to obtain a higher degree of security. This is described in Fig. 3.1.

c) Show that the designed cryptosystem can be used to protect data from different sources with the same and high degree of security. Chapters VI and VII are dedicated to cover this point.

d) Demonstrate that regardless of the plaintext components and nature difference a uniform distribution is obtained in the ciphertext. This phenomenon is demonstrated in Figs. 7.7 to 18. Such figures are contained in chapter VII.

e) Reveal that the chaotic attractor used in a cryptosystem can be concealed to enhance its security properties and resist cryptanalysis as displayed in Figs. 4.17 and 5.7. The previous requirement is complementary to this one by providing resistance to statistical cryptanalysis while this one focuses in giving stealth capabilities to the chaotic attractor. Both of these requirements are of paramount importance for cryptanalysis resistance.

The major contribution of this thesis is a reformulated development of cryptosystems based on cellular automata. Unlike recent previous works in the area, *e.g.*, [AnIS08], [AnSI07], and [ASRI08], which have used cellular automata with very limited characteristics to design cryptosystems. These limited cryptosystems lead to their possible characterization and consequently the possible compromise of data.

1.5 Organization of the Thesis

This thesis is presented with the following structure: (i) four chapters detail the background theory; (ii) four chapters on experimental design, results, and conclusions, and (iii) appendices with details of the FSS10 verification and testing, as well as history of cryptography and alternative cryptographic techniques as is the case of quantum cryptography.

The first background chapter, Ch. 2, introduces the fundamentals of cryptography and CCA. Cryptography is explored ranging from the introduction of the *data encryption standard* (DES) algorithm, which is considered the turning point in the progression from classical to modern cryptography, to the latest advances in cryptography, which are aided by quantum mechanics. Continuous cellular automata capabilities to enter into chaotic regime are analyzed. This achieves the creation of the fundamental basis for the cryptosystem.

The second background chapter, Ch. 3, is the integration of these fundamental building blocks into a complex array that conforms the cryptosystem. The *n*-tuple and encryption/decryption rules allowing the cryptosystem to function are carefully detailed. A detailed implementation of the cryptosystem is described in Fig. 3.1.

Once the cryptosystem framework has been introduced, Ch. 4 presents the verification techniques. Four different verification techniques are considered: (i) FSS10 stationarity, (ii) variance fractal dimension, (iii) spectral fractal dimension, and (iv) surrogate data. The first one explores the capabilities to determine if a signal is stationary towards the strong sense. The second and third techniques provide a quantitatively measure of the fractal complexity of a signal. Lastly, the fourth technique demonstrates that a chaotic attractor can be concealed providing the cryptosystem with stealth capabilities helpful against cryptanalysis attacks.

As final background chapter, Ch. 5 tests the cryptosystem by applying the techniques introduced in the previous chapter. The cryptosystem withstands the four different verification techniques, which are explained in Ch. 4.

The first experimental chapter is Ch. 6. It presents a detailed description of the experiments design. The experimental platform used and the cryptosystem setup is described. The experiments consider a selection of data from different sources that are secured by the cryptosystem.

The experimental results demonstrate that different plaintext sources exhibit uniform histograms once they have been encrypted are discussed in Ch.7. This achievement validates how effective the cryptosystem strength is to resist statistical cryptanalysis.

The conclusions of the research outcomes are described in Ch. 8.

Appendices containing complementary information are also provided. The figures obtained through the verification and testing of the stationarity method FSS10 are included in App. A and B. A historical evolution of cryptography, the art of the secret writing, is included in App. C. An extension of the theory of quantum cryptography is included in App. D.

CHAPTER II

CRYPTOGRAPHY AND

CONTINUOUS-INTERVAL CELLULAR AUTOMATA

A journey across cryptography is provided in this chapter along with its connections with cryptology are brought. The cryptology branching into cryptography and cryptology is explored, which then focus turns onto cryptography. The turning point from the classical to modern cryptography came with the inception of the *data encryption standard* (DES) algorithm [NBS77]. This standard allows to encrypt/decrypt data using a single key, which is known as the first symmetric cryptosystem that was widely used. Various cryptosystems that are used for protecting secrets are also discussed. Finally, the mathematical foundations of CCA are described, which are essential to understand the bases of the cryptosystem that this thesis presents.

2.1 Cryptography Related Definitions

Cryptography and *cryptanalysis* combine to form *cryptology*, which is the science that studies cryptosystems. Cryptography concerns itself with the design of cryptosystems, which is the focal point throughout this thesis, while cryptanalysis studies the breaking of cryptosystems [Tilb03] (Appendix C presents a brief history of cryptography). These terms find their origins in the Greek language. Etymologically, *Crypto* means secret, *Logos* means discourse or study, *Grafos* means writing, and *Analisis* means separation (as opposed to synthesis) [MePS11].

Cryptography is concerned with changing an open message known as *plaintext* (PT) to a

ciphertext (CT) that cannot be deciphered. This process of transformation uses a key ([Ferg10], [KaLi07], and [Bows82]). The scrambled message (a cryptogram) can be either a code or a cipher. A code is a scheme of translating words or symbols from the original message into a new collection of corresponding words or symbols. A cipher is a scheme of translating individual symbols of the original message into another sequence of symbols. The formation of a code or a cipher requires a key. The key is a formula or device by which the code or cipher is created. Encrypting (encoding) is the process of changing a PT to produce a cryptogram. The inverse process of deciphering (decoding) a cryptogram is called decrypting. A set of algorithms for key generation, encryption, and decryption is called a cryptosystem.

A cryptosystem using a single secret key for enciphering and deciphering is known as *symmetric* [AnIS08]. As the number of data receivers in a symmetric cryptosystem increases, the risk of the key being intercepted and decoded also increases. An alternative to reduce this risk is to use two keys: one public key for encryption (known to all receivers), and one secret key for decryption (known to only one receiver) as proposed theoretically earlier by Diffie and Hellman [DiFe76] and implemented in practice by Rivest, Shamir, and Adelman [RiSa78]. The strength of this public-key cryptography is that (i) the deciphering function cannot be derived from the enciphering function due to the methodology of using trap-door one-way functions, and (ii) the associated message signature can verify individually the sender to the receiver.

Cryptosystems should be distinguishable from coding/decoding systems devised for error detection and correction of data either in storage or being transmitted (*e.g.*, [GlDu91], [VaOo89], [Rhee89], [Pete81], and [Wake78]).

Cryptanalysis is concerned with breaking codes and ciphers without any prior knowledge

about the key or algorithms [MePS11], [Swen08], [Sink68], and [Gain56]. This search to find the original message that originated a CT or the key used to cipher it. Cryptanalysis intends to find weaknesses or insecurities in a cryptosystem with for the ultimate purpose of accessing the raw data. Cryptography and cryptanalysis are part of cryptology, which focuses in the study of secrets [MePS11]. Cryptology also includes signal security (*i.e.*, the methods of protecting messages and communications from interception so that they could not be modified, destroyed, disclosed, or compromised in any way) and signal intelligence (the process of obtaining information by intercepting and solving cryptanalysis).

2.2.1 Types of Codes and Ciphers

Simple ciphers include transposition codes such as: message reversal, geometrical patterns, route transposition with many of its variations, columnar and double-columnar transposition (both mono and polyliteral), reciprocal ciphers, mono and polyalphabetic substitutions, decimated alphabet ciphers, digraphic substitutions, random table substitutions, as well as linear and nonlinear scalar and matrix scrambling. For practical implementations of such cryptosystems the reader could be referred to [Schn95]. Cryptosystems protecting secrets by using transposition or substitution techniques can achieve diffusion or confusion of the message respectively.

2.2.2 Data Encryption Standards (DES) & Advanced Encryption Standard (AES)

In 1973, the National Bureau of Standards (NBS), now the National Institute of Standards (NIST), requested the development of a federal standard for computer data protection. In response, IBM developed DES [Bosw82], [Gail78], and [NBS77]. The DES system utilizes

nonlinear ciphering algorithms. It converts 64-bit blocks of PT into 64-bit blocks of CT, using a 56-bit keying parameter whose individual bits are random, and are error protected by 8 odd parity bits. A block of data is subjected to an initial permutation, then to a 16-round computation sequence, and finally to a permutation that is the inverse of the initial one. This DES cryptosystem is symmetric (encryption/decryption is performed by the same key). The DES algorithm represents a turning point from classical to modern cryptography, but it is relatively weak, considering today's easy access to extraordinary computing power to break its symmetric key.

Since the original 56-bit DES can be broken quickly, it is now considered obsolete, and a new symmetric *advanced encryption standard* (AES) was developed in 2001, whose keys offer either 128, or 192, or 256 bits.

2.2.3 Public-Key Cryptography (PKC)

Public-key cryptography was proposed by Diffie and Hellman as an attempt to have privacy and an efficient key distribution scheme that eliminates the need for a secure channel to exchange keys [DiHe76]. Even though that Diffie and Hellman provided no practical implementations of their theoretic proposal for an asymmetric cryptosystem, a base was set to pursue the design of safer cryptosystems incorporating a different and new approach. As a consequence of this idea, asymmetric key cryptography has become a standard for most of the e-Commerce secure transactions providing identification, authentication, authorization, signature, and verification services. The cryptographers that brought Diffie and Hellman ideas to practice are Rivest, Shamir, and Adelman [RiSA78]. An example of a PKC is the Rivest-Shamir-Adelman (RSA) algorithm that uses prime numbers and modular arithmetic for the public and

private keys (*e.g.*, [Gait78], [NBS77], and [RiSA78]). The security of the PKC comes from the difficulty in factoring large composite prime numbers, while it is easy to compute the product of two large prime numbers. The public and private keys are functions of the product of the two primes.

Public key cryptography was also silently developed in the early ‘70s at the United Kingdom Government Communications Headquarters (GCHQ) in Cheltenham before being proposed by Diffie and Hellman [DiHe76] and implemented by Rivest *et al*. [RiSA78]. It was not until 1997 that this classified information was made public and revealed the mathematician James Ellis as the person that started to research the key distribution problem in 1969 [Sing99]. James Ellis collaborated with Clifford Cocks and Malcolm Williamson to work out the fundamental details of PKC [Sing99].

Public key cryptography main strengths are: (i) the asymmetric algorithm is different from the symmetric counterpart. The calculations that support it are based on mathematical functions that demand two keys, one for the encryption process and one for the decryption process; (ii) Any user can generate his/her own pair of keys; (iii) Once a pair of keys are generated, data encrypted with the public key can only be decrypted with the private one; (iv) theoretically it is not possible, except for who calculated his own pair of keys, to obtain the private key once in possession of the public key and vice versa [MePS11] unless a form of cryptanalysis is performed.

2.2.4 Elliptic-Curve Cryptography

Elliptic-curve cryptography, proposed by Victor Miller (IBM) and Neal Koblitz (University of Washington) in 1985 [Kobl95], is an extension of PKC. The security of ECC relies on the

elliptic-curve discrete logarithm problem (ECDLP) in which exponentiation over the discrete prime field is easy, but the inverse (computing the logarithm) is very difficult. For a given key size, ECC offers considerably greater security than RSA in the sense that a small ECC key is comparable to a bigger one in RSA. For example, for 156-bit key size in ECC requires an equivalent 1024-bit RSA key. Other equivalent ECC to RSA ratios are 256/3072, 384/7680, 512/15360. Elliptic-Curve Cryptography was initially developed for devices with limited memory, power, or computing capabilities like smartcards and electronic devices running on limited microprocessors. One remarkable feature that has made ECC extensively used is that most of its algorithms are not patented.

An elliptic curve in Cartesian coordinates is the set of solutions (x, y) to an equation of the form (*e.g.*, [Kobl94], [Mene93], [Rosi99], and [Wash08])

$$y^2 = x^3 + a_1 x + a_2 \tag{2.1}$$

together with an extra point O which is called the point at infinity. For applications in cryptography, only a Galois finite field of q elements, $\mathrm{GF}(q)$, is considered. When q is a prime, one can think of $\mathrm{GF}(q)$ as the integers modulo q. For a given pair of numbers, the forward computation is simple, but its inverse is practically intractable at this time.

Public key cryptosystems for ECC are analogues of cryptosystems available for other discrete logarithm based systems (such as the multiplicative group of a finite field), including Diffie-Hellman key exchange, ElGamal public key encryption, and ECDSA (an analogue of the US government's digital signature standard). In 2005, the National Security Agency (NSA)

decided to adopt elliptic curve based public key cryptography. Research has recently been shifting from elliptic curves to lattices [Gent09], [MiRe08], and [Rege06].

2.2.5 Quantum Cryptography

In 1984, *quantum cryptography* (QC) was introduced by Bennett and Brassard [BrBe84]. Quantum cryptography ensures the confidentiality of the information transmitted using the *Heisenberg uncertainty principle* as its basis. By the uncertainty principle, an eavesdropper cannot know everything about an elementary particle carrying a key bit, and part of the compromised data can be destroyed because measurement is destructive in quantum mechanics. This feature allows detecting eavesdroppers. The most mature application of QC is the distribution of secret keys. However, a source of truly random numbers (*i.e.*, noise in a resistor) and a classical authenticated channel are needed in the creation of these secret keys. A more detailed description is included in appendix D.

2.2 Continuous-Interval Cellular Automata (CCA) and Chaos Phenomena

Cellular automata are stylized, synthetic universes defined by simple rules much like those of a board game (*i.e.*, chess is a game with simple rules, but its evolution during a match unpredictable). They have their own kind of matter, which whirls around in a space and a time of their own. An astounding variety of CA exists. One can actually recreate CA or design new and watch them evolve. A cellular automata machine is a universe synthesizer. Like an organ, it has keys and stops by which the resources of the instrument can be called into action, combined, and reconfigured. When cellular automata are configured in a computer monitor, its color screen is a window through which one can watch the universe that is being “played” [ToMa87].

The initial developments in *cellular automata* (CA) are based on the work of Konrad Zuse, Stanislaw Ulam, and John von Neumann. Cellular automata were suggested by Stanislaw Ulam to John von Newmann in the 1940s to provide a more realistic model for the behaviour of complex extended systems. The German, Konrad Zuse, while isolated in a mountain peak in Austria hiding from the Nazis, conceived ideas related to high-level programming languages and "computing spaces" as CA [ToMa87]. Cellular automata are closely related to the first computing machines [PeJS04]. Stephen Wolfram has prompted another revival of CA [Wolf02]. A cellular automaton is a *finite state machine* (FSM) whose state changes in discrete steps [PeJS04]. Given a set of initial conditions and a set of rules, a CA produces a sequence by its inherently growing nature. According to Hachtel and Somenzi [HaSo96], any sequential circuit can be modeled as a FSM, consisting of combinational logic and memory.

Cellular automata are discrete dynamical systems whose behaviour is completely determined in terms of a local relation, much as is the case for a large class of continuous dynamical systems defined by *partial differential equations* (PDE). In the stylized universe of a CA, space is defined by a uniform grid with each cell containing a few bits of data. Its time advances in discrete steps. The laws of the CA universe are expressed by a single procedure through which at each step each cell computes its new state relating to its neighbours. Thus, the system's laws are local and uniform (the same everywhere). When we change the CA initial conditions we can get a different sequence (history) and when the law is changed a new set of dynamics (a new universe) is created. Cellular automata are capable of creating self-contained systems that require no interaction with external components of a computer, which is more helpful than *transducers* (systems producing a steady output stream of information in response to a steady input stream)

[ToMa87].

Although a CA can be defined by a very simple program, it is capable of producing behaviour of great complexity, or chaotic, and it is sensitive to small changes in its initial conditions [Wolf02]. As mentioned in the introduction, *continuous-interval cellular automata* (CCA) are generalization of CA in which a cell can have any value from the continuous interval $[0, 1]$ [Wolf02]. This interval extends the set of possible values in which the behaviour of a CCA can be expressed. Continuous cellular automata remove some of the discreteness existing in ordinary CA [Wolf02].

Dynamical systems theory studies the emergence of well-characterized collective phenomena (*e.g.*, ordering, turbulence, chaos, symmetry-breaking, and fractality) in systems consisting of a large number of individuals connected by nonlinear couplings. Cellular automata provide a rich and continually growing collection of representative models where these phenomena can be isolated and studied. Physical modeling has gained benefit from CA in areas like reversibility, diffusion and equilibrium, physical dynamics, collective phenomena, and ballistic computations among others [ToMa87].

The behaviour of a CA can be separated into four classes: (i) simple behaviour in which the CA arrives at the same final state, (ii) distinct possible final states, but all consisting of simple periodic structures, (iii) complex behaviour, often random or chaotic, and (iv) fluctuation between order and randomness [Wolf02].

According to the previous paragraph, this thesis considers CCA capable of exhibiting behaviour of Class 3, which can yield random sequences or patterns that never repeat (*i.e.*, chaotic). This highly complex behaviour achieved by CCA is defined by simple laws or rules.

Intrinsic randomness is a phenomenon in which a system does not require external factors as the source for randomness. Instead, the source of randomness is embedded in the system itself, as is the case for some CCA. The term embedded refers to a computer system, including the source of randomness, encrypter/decrypter, and key generation modules, in a VLSI chip. Intrinsic randomness generation puts all the components in a system to work in producing new randomness [Wolf02]. These reasons make CCA highly amenable to cryptography.

The rounding errors process, either by the ceiling or flooring operator, in computing is due to the finite resolution in computers. Certain simple rules in CCA that are defined by the computation of basic arithmetic operations make them capable of exhibiting chaos. Very often the terms *dynamical* and *dynamic systems* are used as synonyms in the literature. It is important to point that there is a difference between *dynamic* and *dynamical systems* in terms of behaviour. The former ones could exhibit cycle stability while the latter ones are also capable of entering into a *chaotic regime* [Kins11a, Ott02 and Stro00]. The route to chaos in dynamical systems exhibits cycle stability and period doubling (*i.e.*, bifurcation). The number of points present in a cycle is composed of powers of two [Kins11a]. A graphical representation of this chaotic nature is provided in Figs. 2.1 to 2.3 by a unicellular CCA that is described in the next chapter.

According to the canonical definition of chaos by Devaney ([Deva92] and [Kulc08]), the main characteristics of chaotic behaviour are (i) topological transitivity, (ii) density of periodic points, and (iii) sensitive dependence on initial conditions. The last two features are implied by the first one making it the sufficient condition for a dynamical system to develop chaos, and the third one ensures that the trajectories in a chaotic signal vary greatly for different small values on its initial conditions [Kins11a]. By varying the initial conditions, a totally uncorrelated sequence

can be obtained, and as a consequence the number of available waveforms could be infinite [WLZF01].

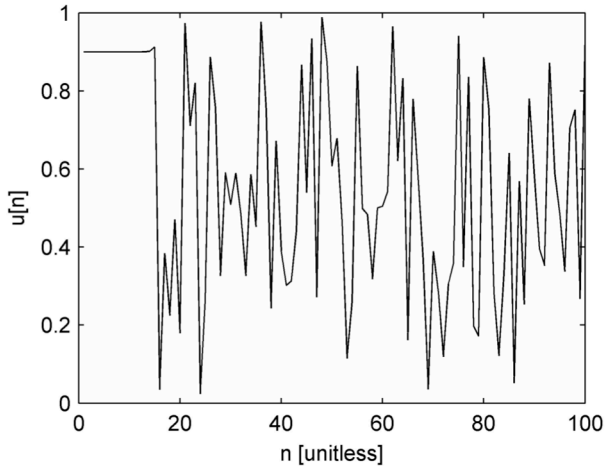


Fig. 2.1. One-cycle stability prior to the chaotic sequence. Initial conditions at $u[0]=0.9$ and parameter $g=11$.

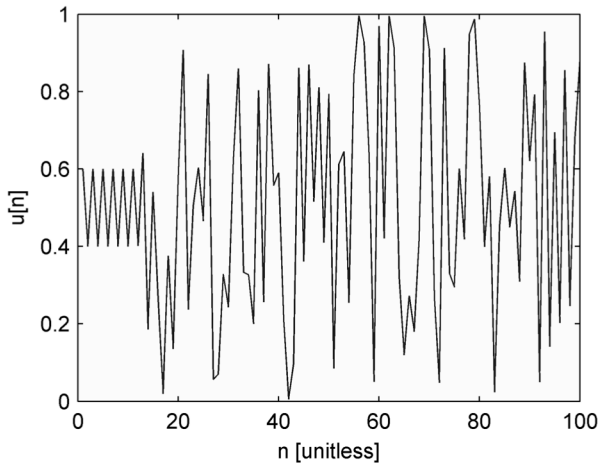


Fig. 2.2. Two-cycle stability prior to the chaotic sequence. Initial conditions at $u[0]=0.6$ and the parameter $g=19$.

A rule that makes a one-dimensional CCA exhibit chaos is defined by

$$u[n] = g\left(u[n-1]\right) - \left\lfloor g\left(u[n-1]\right)\right\rfloor, \text{ for } n = 1,\ 2,\ \ldots \tag{2.2}$$

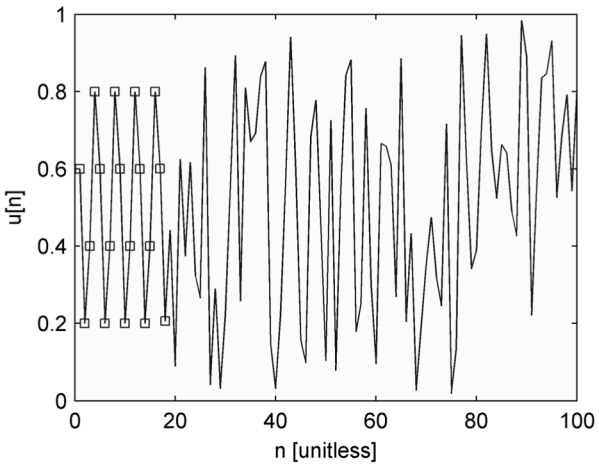


Fig. 2.3. Four-point stability prior to the chaotic sequence. The squares are included as visual aid to identify the points in the four-cycle. Initial conditions at $u[0] = 0.6$ and the parameter $g = 7$.

where g is a real parameter, $g \in \mathbb{R}$, the notation $u = [\bullet]$ signifies a discrete map, and the symbol $\lfloor \bullet \rfloor$ denotes the floor operator. The initial condition is given at $n = 0$, and just the fractional value is kept at each step. This rule is optimized for a unicellular CCA to achieve one of the fastest possible ways to compute a chaotic sequence as it only takes one multiplication and one subtraction. The dynamical system described by (2.2) produces a strange attractor whose points appear in an array of "needles" in a three-dimensional pseudo-phase space. The proof of the equation (2.2) having a chaotic attractor is addressed by experimental demonstration. This experimental proof considers different initial conditions for equation (2.2) and observing that the chaotic attractor always emerges from the sequences obtained. This experimental proof follows the *delay embedding theorem* proposed by Floris Takens [Take81], which sustains the reconstruction of the strange attractor shown in Fig. 2.4. For this thesis, an attempt to proof that equation (2.2) is indeed chaotic from the mathematical point of view has not been done. When

the parameter g takes odd valued integers, a tilted cube of $g \times g$ needles is visible in the pseudo-phase space, and all the needles are of the same length, as illustrated in Fig. 2.4.

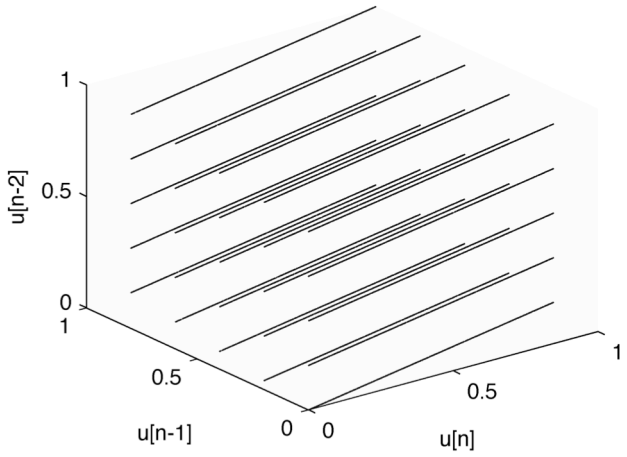


Fig. 2.4. Multi needle strange attractor in the pseudo-phase space.
Parameter $g = 5$.

The parameter g can have fractional values, but when this happens some of the needles can vary in length. As the parameter g increases, the array of needles becomes denser in the pseudo-phase space. For some values of g, the multi-needle attractor exhibits features like point stability and period doubling, which is a characteristic of chaotic phenomena, and then a transition into a chaotic regime, as illustrated in Figs. 2.1 to 2.3. As seen in the figures, the periods of stability just prior to the chaotic sequences occur when the initial conditions are composed of one digit in the fractional value. If the fractional value includes more digits and is not a negative power of two, then the final sequence value converges to zero. Table 2 lists the peculiarities for the *multi-needle attractor* proposed by this thesis, which is one of the fastest chaotic attractors as measured by its number of operations. In general, continuous models (*i.e.*, flows) of chaotic systems based on DEs whose solutions involve more computations than chaotic systems based on CA. For

example, the three-dimensional chaotic weather model with its butterfly effect as developed by Lorenz [Lore63] is a set of coupled DEs.

TABLE 2.1: Multi Needle Strange Attractor Parameters.

REGIME	INITIAL CONDITION VALUE	PARAMETER g VALUE
Extinction	All integers and fractional values	Even integers or values less than one
One Point Stability at the Beginning	Fractional value of one digit that is not a negative power of two	Integers ending in one except number one
Two Point Stability at the Beginning	Fractional value of one digit that is not a negative power of two	Integers ending in nine
Four Point Stability at the Beginning	Fractional value of one digit that is not a negative power of two	Integers ending in three or seven
Chaos	Fractional value of more than one digit that is not a negative power of two	$g > 1$ not being an even integer. g can be a rational value.

It is possible to extend the idea from the unicellular multi-needle attractor to a CCA that has more cells in a n-dimensional system, which averages the summation of the neighbour cells and the current cell. In this model,

$$u[n] = \frac{g\sum_{i=-1}^{1}\left(u[i,n-1]\right)}{3} - \left\lfloor \frac{g\sum_{i=-1}^{1}\left(u[i,n-1]\right)}{3} \right\rfloor \tag{2.3}$$

the right and left neighbours of a central cell are added and averaged, and just the fractional value is kept. Since the model provided by (2.3) requires more operations for its computation in comparison with the CCA depicted by (2.2), the focus in this thesis is the latter.

Chaotic phenomena have been widely used in different technology related areas. Digital communications that uses broadband signals is a field that also researches chaos models. A chaotic signal produces a wideband noise-like signal with robust and reproducible statistical properties. Due to its wideband nature, a modulation scheme using chaotic basis functions is potentially more resistant to multipath propagation than one based on sinusoids. One factor which limits the performance of all telecommunications systems is interference. In conventional systems based on periodic carrier signals, where the *spread spectrum code division multiple access* (SS-CDMA) technique is not used, signals can be made orthogonal by putting them in different frequency bands [*frequency division multiple access* (FDMA)] or time slots [*time division multiple access* (TDMA)], by ensuring that the basis functions are orthogonal to each other (*i.e.*, using sine and cosine basis functions), or by using orthogonal electromagnetic polarization. If these requirements are not met, interference occurs [KeRS00].

In contrast with periodic signals, chaotic signals decorrelate rapidly with themselves. Also, chaotic signals generated by different chaotic circuits are almost orthogonal. This means that the correlation, equivalently the interference, between two chaotic signals that are generated by

unsynchronized chaotic circuits started from different initial conditions and/or having different circuit parameters, is low [KeRS00]

2.3 Summary

The definitions related with cryptology have been provided. The cryptology branching into cryptography and cryptanalysis has been stated and different cryptographic technologies have been described in detail. Also, this excursion includes, the turning point from the classical to modern cryptography through the inception of digital symmetric cryptosystems. The foundation of asymmetric cryptosystems using prime numbers and elliptic curves is also explored. Ultimately, the theoretical bases of the proposed cryptosystem are extensively described. The compact implementation of this cryptosystem based on CCA is meticulously described in the following chapter. The cryptosystem implementation relies on using the dynamical system based on CCA presented here. The last two paragraphs provide insight into the practical uses of chaos phenomena in broadband digital communications.

CHAPTER III

CRYPTOSYSTEMS BASED ON CELLULAR AUTOMATA

3.1 The First Reported Cryptosystem Based on Cellular Automata

The first known cryptosystem based on CA was proposed by Stephen Wolfram in 1985 [Wolf86]. The cryptosystem is based on a stream ciphering technique described by

$$u[n,o]=u[n-1,o-1]\oplus\left(u[n,o-1]\,|\,[n+1,o-1]\right) \tag{3.1}$$

where $\oplus$ stands for the *exclusive or* operator and $|$ for the *or* operator. The indexes n and o denote the relative position of one cell to another and the evolution state of the cellular automaton respectively. This cryptosystem is not continuous. It provides the binary states zero and one at any stage of its evolution for its cells. The initial state of the register is used as a seed or key. Ciphertext then is obtained *xor*-ing the binary plaintext with $u[n]$. The plaintext can be recovered repeating the same operation, but only if $u[n]$ is known. Figure 3.1 shows the pattern of cell values produced by (3.1) with a seed consisting of a single nonzero cell. This figure shows an array of 128 by 64 cells. It is seen that complex behaviour emerges even when the initial conditions are too simple as just one cell having a value of one and the rest of them are kept with a value of zero. These initial conditions can be seen in the top row in Fig. 3.1.

In 2002, Stephen Wolfram refers to the CA described by (3.1) as *rule 30* [Wolf02]. The equation (3.1) can also be expressed by the Boolean function

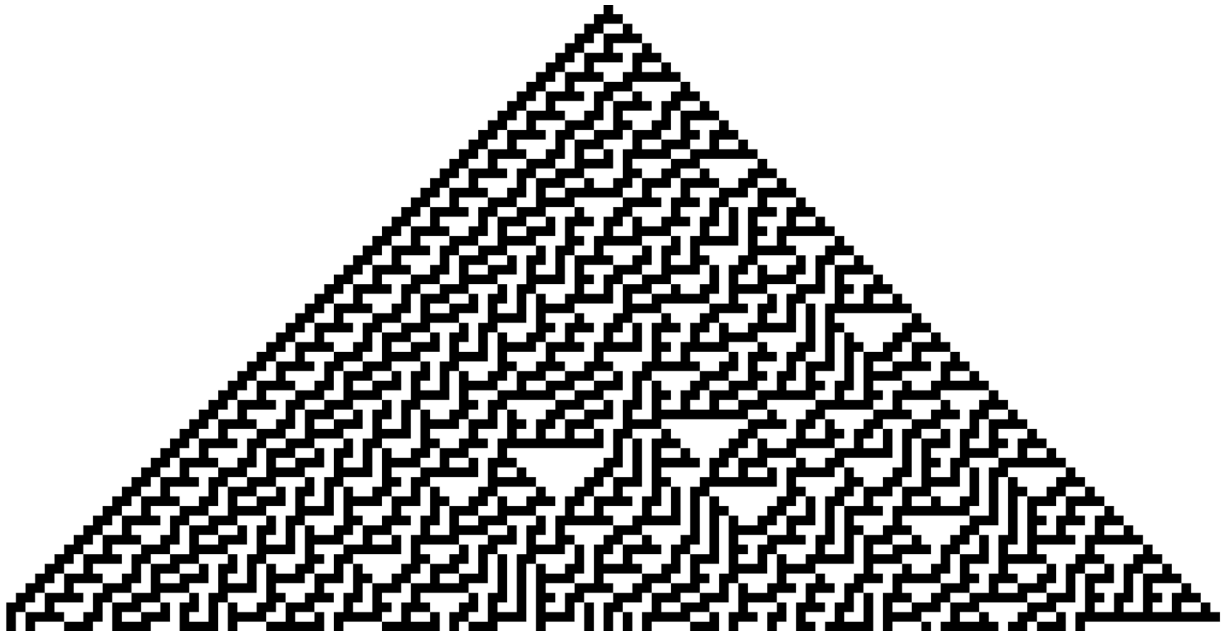


Fig. 3.1. Cellular automaton obtained using the rule 30. The picture shows an array of 128 by 64 cells. Each row denotes a given state in the automaton evolution. As shown in the first row, only a single cell has a value of one.

$$\begin{aligned} &f[111]=0, f[110]=0, f[101]=0, f[100]=1 \\ &f[011]=1, f[010]=1, f[001]=1, f[000]=0 \end{aligned} \tag{3.2}$$

Where the numbers inside the brackets are in binary format. The intrinsic randomness found in this CA is so rich that the Mathematica simulation software uses it as a *pseudorandom number generator* (PRNG). Wolfram refers to the pattern of this automaton as random *in many respects*. However, he provides no complexity measures of this CA. This thesis looks at the use of the VFDT and SFD as mathematical aids to measure the complexity of the behaviour of dynamical systems that can be constructed using CA. This approach considers a quantitative perspective of the pattern in the evolution of CA. The outcome of the measuring approach through polyscale techniques, which are rooted in fractal analysis, provides a specific value that characterizes unequivocally the behaviour of a CA. The importance of the polyscale techniques used herein is

that they can be applied to cell state streams from any CA.

3.2 Correlation Attacks Against Cryptosystems based on Cellular Automata

The fast correlation attack described by Meier and Staffelbach [Penz96] works well against certain classes of stream ciphers that are based on *linear feedback shift registers* (LFSR). One of the requirements is that the number of taps of the characteristic polynomial must be small (normally less than ten). An algorithm developed by Walter Penzhorn [Penz96] uses the theory of cyclic block error-correcting codes to cryptanalyze stream ciphers with a higher number of taps as long as 100 bits.

The importance of this attack is that it is helpful in the cryptanalysis of the cryptosystem described in the previous subsection that is based on the *rule 30* [MeSt92]. The Meier-Staffelbach attack against *rule 30* exploits the central column displayed in Fig. 3.1 and its partial linear properties [MeSt92]. These features are leveraged by observing the output sequence that lead to the reconstruction of sections of the internal evolution of the *rule 30* CA [MeSt92]. The CA is then reconstructed backwards. Cryptanalytic systems that use dedicated hardware can potentially compromise cryptosystems of 1,000 cells.

Chaotic phenomena in CA have not yet been studied in depth. Cellular automata based cryptosystems that are designed to exploit consciously the chaotic behaviour have not been proposed in the known literature until now, through the presentation of this research. Previous cryptosystems based on CA are weaker than rule 30 and consequently are easier to cryptanalyze (*e.g.*, [AnIS08], [AnSI07], and [ASRI08]).

Programmable cellular automata (PCA) consider more than one rule for each cell [NaKC94]. Recently, implementations of PCA considering three (*i.e.*, [Angh11] that combines

rules 51, 60, and 102) and four different rules (*i.e.*, [ToPe01]) for each cell. However, the chaotic behaviour in CA has not been fully exploited or even measured. This thesis proposes the first cryptosystem ever that consciously exploits the chaotic behaviour in CA. Polyscale techniques are used to measure the complexity of its chaotic phenomena defining a quantitative characterization. The cryptosystem proposed throughout this thesis directly exploits and harness chaotic phenomena, exceeding the recent developments reported in the literature. The following subsection provides an in depth insight about the cryptosystem.

3.3 Multi Modular Dynamical Cryptosystem

Continuous-interval cellular automata capable of exhibiting chaos are highly valuable in cryptography because this approach allows for a theoretical infinite number of possible keys in the keyspace. As defined in Chapter II section 1, cryptographic techniques are divided into symmetric-key (when the same key is used for the encryption and decryption processes), and asymmetric-key (when one key is kept private and one or more keys are released publicly) [AnIS08]. The sensitive dependence for slight changes in the initial conditions of dynamical systems allows the design of symmetric cryptosystems capable of generating a great number of keys, theoretically infinite, for the encryption and decryption processes.

A proposed cryptosystem model based on chaotic CCA is illustrated in Fig. 3.2. “Modular” denotes the interaction between the different modules, each of them containing a dynamical system based on CCA to obtain a final sequence useful in encryption/decryption. The encryption/decryption processes request sequences from the *sequence generator* (SG), which acts as a master. The sequence generator replies with a *sequence S* of *length L* by chaotically mixing small *contribution sequences C* also chaotically generated. Each sequence C is provided

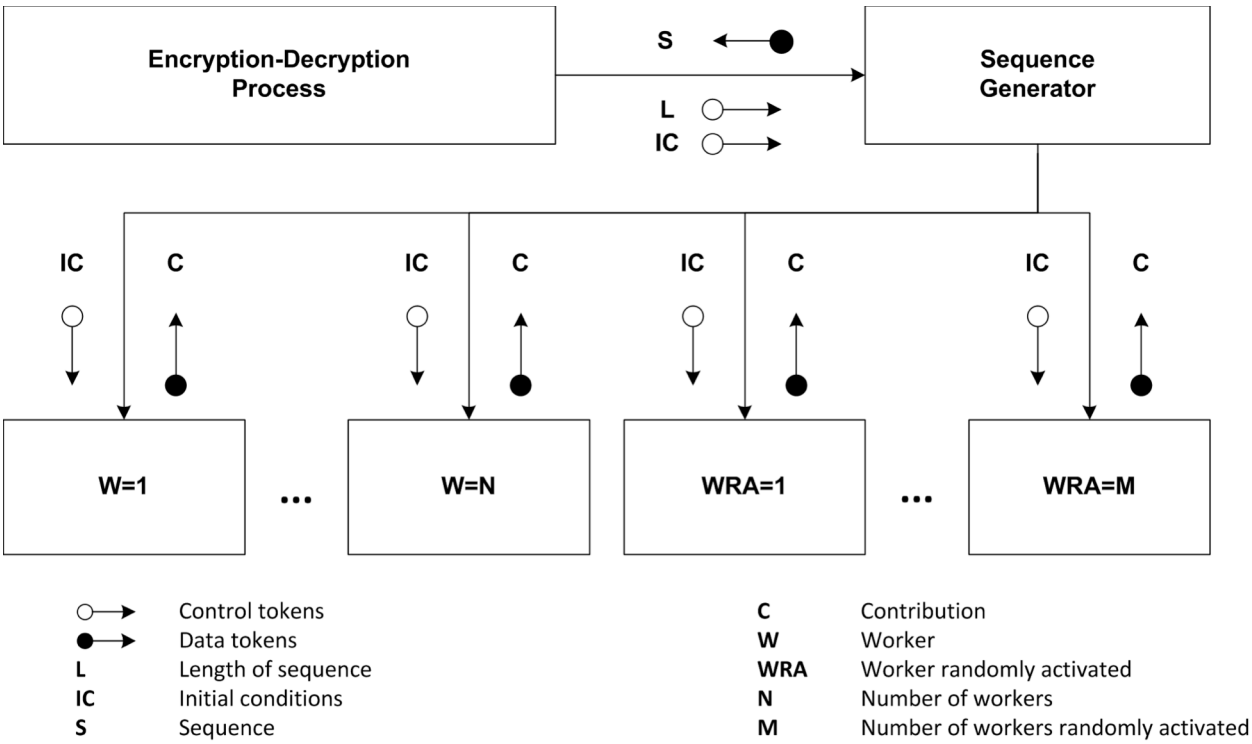


Fig. 3.2. Multi-Modular dynamical cryptosystem structured chart.

by the set of *workers* (W) and also by the set of *workers chaotically activated* (WRA) in which a group of them are activated chaotically at each step. The initial conditions of both sets are provided by the SG. The initial conditions used by the W and WRA are never equal given the chaotic regime present in the SG. An instance of the chaotic CCA, which is defined previously by (2.2), is contained in the SG, the W, and the WRA, with different initial conditions and parameters. The sequences C also vary in length chaotically. The sequence generation process stops once the sequence S has reached the specified length.

As a characteristic of chaotic processes, the variation of the initial conditions in the W and WRA by the SG never exhibits periodicity. A process of chaotically mixing chaotic sequences takes place in the SG, with the benefit of achieving higher secure characteristics in the encryption/decryption processes. Other recent approaches (*e.g.*, [ASRI08], [AnSI07], [PeZL09],

[YiRY09], and [AnIS08]) have limited considerations either for the manner of how CA are used, or the type of data that they are capable of protecting. It is possible to encapsulate any dynamical system like the well known Lorenz [Lore63], Rössler [Röss77], and Hénon [Héno76] attractors in a given module in Fig. 3.2. Since those dynamical systems require more mathematical operations in their implementations compared with the *multi-needle attractor* herein introduced, they are computationally more costly. Due to this reason of being costly such dynamical systems are not considered in this thesis.

The cryptosystem explained above is described by the following six-tuple

$$\mathcal{T}=(\mathbb{P},\ \mathbb{A},\ \mathbb{C},\ \mathbb{K},\ \mathbb{E},\ \mathbb{D}) \tag{3.2}$$

where $p \in \mathbb{P}$ represents general plaintexts from different information sources, $a \in \mathbb{A}$ restricts all the CCA used in the cryptosystem modules to exhibit chaos, $c \in \mathbb{C}$ is the set of ciphertexts, $k \in \mathbb{K}$ is a set of possible keys, $e \in \mathbb{E}$ is the space of possible encryption rules, and $d \in \mathbb{D}$ is the space of possible decryption rules. Notice that according to Stinson [Stin06], a cryptographic system is composed of a quintuple $\mathcal{T}_S=(\mathbb{P},\ \mathbb{C},\ \mathbb{K},\ \mathbb{E},\ \mathbb{D})$. However, since the CA are of paramount importance for this chaos-based cryptosystem, the term $\mathbb{A}$ is included to indicate that the CCA utilized are capable of exhibiting chaos. The elements in the sets $\mathbb{P}$ and $\mathbb{C}$ are considered as unique, and all the sets in $\mathcal{T}$ are proper.

The encryption and decryption keys used are defined as

$$e_k[n] = \left(p[n] + u[n]\right) \bmod r \tag{3.3}$$

$$d_k[n] = \left(p[n] - u[n]\right) \bmod r \tag{3.4}$$

where $p[n]$ represents the original data sequence, $u[n]$ is the encryption/decryption sequence generated by the SG, and r is the dynamical range that lets a representation of $p[n]$ where no data are lost and therefore achieves perfect encryption/decryption.

The subsequent subsection presents an enhancement to the cryptosystem core to counteract possible signal intelligence operations. This aims to counter possible attacks that exploit: (i) identification of the chaotic system in the time domain by its state variables, (ii) using an approximate model that could lead to recover the message, and (iii) finding the synchronization pattern between the encrypter/decrypter modules.

3.4 Enhancement of the Cryptosystem Using Surrogate Data

In order to counteract the known apparent weaknesses of cryptosystems based on phenomena (stated in the final paragraph of the previous section) a method that strengthens them tremendously using surrogate data to conceal chaotic attractors, which are the target when attackers attempt to make sense of encrypted data using chaos, gives the CCA based cryptosystem an outstanding advantage. This feature provides stealth capabilities to the chaotic attractor, which makes the cryptosystem robust against cryptanalysis by signal intelligence. It is important to mention that the CCA based cryptosystem presented in this research is invulnerable

to the known attacks against cryptosystems based on chaotic systems. This invulnerability is achieved by (i) avoiding using synchronization and (ii) providing the cryptosystem chaotic attractor with stealth capabilities by means of concealment using surrogate data. Surrogate data has never been used to conceal a chaotic attractor in the past, and then a novelty is presented in this thesis. This unique approach makes this CCA based cryptosystem even stronger. It is then *the most secure cryptosystem based on chaotic phenomena* that has been formally tested in the reported literature.

The fact that cryptanalysis has the ultimate goal of breaking the ciphertext and getting to the data that a cryptosystem secures, should not be ignored when designing secure communications systems. This research approaches cryptosystems design very seriously and a conscious effort to eliminate possible flaws is concurrently present.

Surrogate data can conceal a chaotic attractor. This research considers two different types of surrogates to conceal the chaotic attractor: (i) random shuffling and (ii) Fourier based. Random shuffling surrogates have been selected because they destroy the correlation of the temporal order between samples in a given signal or stream of data. Another advantage of random shuffling surrogates is that all statistical moments are not modified, and hence they do not alter the white noise characteristics of the cryptosystem, which is highly advantageous when protecting data against statistical cryptanalysis. Fourier based surrogates are not selected because they have drawback of the possible change of the plaintext due to rounding errors in the transformation between the time and the frequency domains.

The theoretical background of the surrogate data method used to strengthen this cryptosystem is further described in the chapters IV and V sections four and five respectively.

Chapter IV verifies the surrogate data selected with a known dynamical system in the chaotic regime (*i.e.*, Hénon map) while chapter V presents the testing of the cryptosystem chaotic attractor. In both cases, the Hénon map and the multi-needle attractor no trace is left for an attacker to conclude that the time series analysed belong to chaotic processes.

3.5 Summary

A comprehensive description from the beginning of cellular automata based cryptosystems by rule 30 to the most complex cryptosystem based on continuous cellular automata has been given. The cryptosystem robustness to resist a wide variety of attacks such as (i) statistical (*i.e.*, frequency and correlation), (ii) identification of the chaotic system in the time domain, (iii) using an approximate chaotic attractor model, and (iv) finding the synchronization patterns. The cryptosystem model has been discussed in depth. Finally an enhancement to increase the already robust degree of security of the cryptosystem has been examined. The tools used to test the cryptosystem are verified in the following chapter.

CHAPTER IV

VERIFICATION

It is always necessary to verify that the testing tools are properly designed as well as functioning as they are intended to. This process is the *verification* of the testing tools. When designed properly the testing tools provide satisfactory solutions to known inputs. The inputs and outputs sets are *a priori* known. Verification is important to ensure that an algorithm implementation is reliable either in hardware or software. It allows for operational certainty of the testing tools when applied in a given area (*e.g.*, mathematical analysis, signal processing, and control in industrial environments).

A strong significance is achieved when after verifying that the testing tools provide right outcomes because then it is useful to test the cryptosystem. A cryptosystem provides data security in its applications, being imperative to its users that it is effective. Thus, a rigorous testing of its security degree is necessary. Codebreaking using cryptanalysis is the most important aspect of secret intelligence around the globe. Cryptology by means of cryptanalysis produces much more trustworthy information than spies do. The information provided by cryptanalysis even influences governmental policies [Kahn67]. Therefore, verification tools should be carefully designed when used in cryptosystems.

4.1 Stationarity FSS10

Formally, a time series $\{X_t\}$ where $t \in \mathbb{N}$ is considered *strongly stationary* if for any set of times $t_1, t_2, ..., t_n$ and any integer k the joint *probability distributions functions* (pdf) of

$\{X_{t_1}, X_{t_2}, ..., X_{t_n}\}$ and of $\{X_{t_1+k}, X_{t_2+k}, ..., X_{t_n+k}\}$ coincide. Stationarity is a characteristic of the process, from which a signal under analysis is obtained, itself [WiKP98]. This time independency notion complies *within an observation period* in which the statistical characteristics of the signal do not vary [KaSc04].

Detecting stationarity in a time series is not an obvious and trivial task. It is important to note that it is never possible to truly establish strong stationarity in experimental data since either the time independence of an infinite number of central or non-central moments $M_k^{t_1 t_2 \cdots t_k}(t) = \langle (X_{t+t_1} - \mu)(X_{t+t_2} - \mu)...(X_{t+t_k} - \mu) \rangle$ has to be tested [WiKP98]. Previous research has attempted to provide helpful analytical tools to detect stationarity in time series. An example of this is a modified χ^2 test proposed by the seminal work developed by Witt, Kurths, and Pikovsky [WiKP98].

The realization of a time series (*e.g.*, autoregressive process, fractional Brownian motion, hearth rate variability, or chaotic signals) is *stationary* if its essential statistical properties, moments, are time independent. Two types of stationarity exist: *strong* (SSS) and *weak* sense (WSS). In the last case, only the first two statistical moments are time independent [WiKP98]. However, both notions of stationarity are connected to the reproducibility of time series measurements. In general, a measurement of any kind is more useful the more reproducible it is. It is necessary to know that the measured numbers correspond to properties of the studied object, including some margin for measurement error. Similarly, for time series measurements, reproducibility is closely connected to stationarity [KaSc04].

It is important that a cryptosystem can resist cryptanalysis by means of statistical methods.

Therefore, it is necessary to use a testing tool to formally determine if the cryptosystem is stationary. Verifying that this tool is reliable by providing correct results using a time series having known properties (*i.e.*, Uniform distribution) serves as a basis to determine whether the cryptosystem presented is not vulnerable to statistical cryptanalysis. It is important to note that if a cryptosystem is found to be non-stationary it is flawed and easily exploitable, therefore unsafe. This vulnerability is due to the possible presence of *hooks* in the ciphertexts produced by a non-stationary cryptosystem.

The assessment approach to determine the cryptosystem stationarity is to construct an analysis tool capable of testing if a time series, in general, is stationary. The preferred features in this test are: (i) that it could work even having just a few thousand elements (data scarcity) in the time series under analysis and (ii) that it could test for a stronger demand than just *weak sense stationarity* (WSS). A time series under analysis should cover a stretch of time much longer than the longest characteristic time scale that is relevant for the evolution of the system [KaSc04]. The justification to develop a tool capable of determining if a time series is stationary under higher requirements than those of WSS, bringing a sustainable basis, is to clear assumptions.

A stationarity test proposed in the literature [WiKP98] divides a realization $\{x_t\}_{t=1}^{n}$ of the system into l non-overlapping windows $\left\{x_\tau^j\right\}$ with equal length n_w using

$$\left\{x_\tau^j\right\} = x_{(j-1)n_w+\tau},\ j = 1 \ldots l,\ \tau \ldots n_w \tag{4.1}$$

This procedure produces an artificial data *set* in which statistical tools can be used. In the work

presented here, l is desired to have a minimum value of 30 for statistical significance. A critical consideration is to determine a proper size of the window x_τ^j. There is no general method reported in the literature so far to determine a window size to divide a given time series for stationarity analysis. An equation reported in the literature by [WiKP98] assists empirically in selecting the appropriate window length l:

$$f_{SL}(n,n_p)=\frac{a(n)+b(n)\ln(n_p)}{n^c} \tag{4.2}$$

The value f_{SL}, or significance level, from this equation can be used as the window length l. This equation determines the significance level for an arbitrary sequence length n. The parameters used in this function are defined as: $a(n)=a_0+a_1\ln(n)$, $b(n)=b_0 n^{b_1}$, $n_p=0.15n$ corresponding to the length of the autocorrelation function, $a_0=1.25$, $a_1=-0.078$, $b_o=0.340$, $b_1=-0.157$, and $c=0.36$ [WiKP98]. It is seen in Fig. 4.1 that approximately from a value of

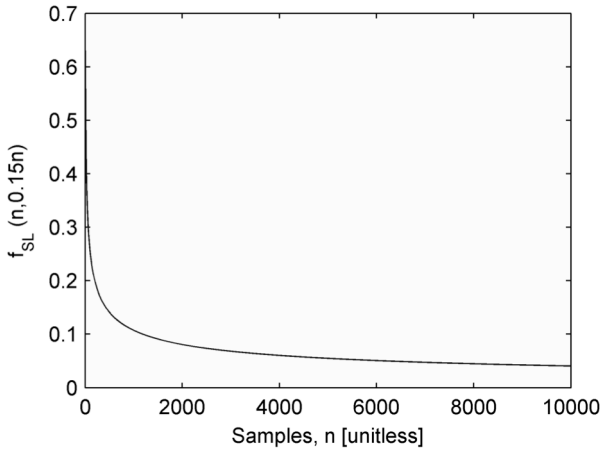


Fig. 4.1. Level of significance $f_{SL}(n,n_p)$ over sequence length n.

greater than $2{,}000$ samples the significance level remains fairly constant for a sequence length $n = 10{,}000$. It is expected that the value of f_{SL} converges to zero when n approaches infinity. The length of the autocorrelation function is chosen as $n_p = 0.15n$. The approach to determine the window length l based on the equation just described is brought here to show how complicated and difficult it could be to solve empirically some of the problems related with stationarity, particularly the window size. However, this equation does not form part of the FSS10 method used as a basis for testing for stationarity in this thesis. The verification of the FSS10 method is now described, opening in the following paragraph.

Testing for strong stationarity requires the evaluation of an infinite number of moments. Therefore, it is unfeasible to determine a conclusion about SSS stationarity in a time series given the finite memory, computing power, and time limitations in computers. The growing error in calculations of higher-order statistical moments, caused by power calculations, cannot resemble the true nature of the process given the finite resolution in computers. Once this computational error exceeds the resolution range in a computer, the value of the statistical moments no longer resembles the process under analysis. Therefore, a threshold in the number of statistical moments that are calculated is essential to avoid this computational error. The threshold set for the number of statistical moments employed to assess stationarity is 10. This stationarity evaluation is consistently addressed throughout this thesis as *Finite Sense Stationarity 10* or *FSS10* for short, which is a new framework to assess stationarity introduced in the literature by this research. Stationarity 10 surpasses the requirements of *weak stationarity* (WSS). Stationarity 10 can provide a graphical representation to analyze if the time series under test is time independent. Stationarity 10 contributes a measure of stationary towards the *strong sense* (SSS).

Consequently, testing of FSS10 can determine the existence of an invariant ergodic measure. The *finite sense stationarity* could have an arbitrary number of statistical moments (*e.g.*, FSS7 if seven statistical moments are considered). However, three constrains exist when implementing it: (i) the growing error, (ii) the computational power required, and (iii) the vanishing moments. These constrains are challenging when considering more than 10 moments in an implementation of *finite sense stationarity*.

The method proposed in this thesis to assess whether a time series is FSS10 stationary divides the time series in 30 segments or windows for statistical significance. For the comparison of the pdfs from the window i^{th} to the j^{th}, the evaluation of the 0^{th} to the 10^{th} moment is then performed.

The stationarity test FSS10 analyzes different window lengths to determine whether a signal is stationary. The biggest window length considered is $n_w = 2^{13}$ and the minimal window length is set at $n_w = 2^5$. Windows of smaller lengths are not explored because the minimal statistical significance does not hold. The number of non-overlapping windows $l = 30$ allows graphical comparison for a set of statistical moments that goes from the zeroth to the tenth. The purpose of including the zeroth statistical moment is to establish a pre verification of the analysis tool. By definition, the zeroth statistical moment always has a value of one in any pdf form. This allows detection of obvious errors during the implementation phase of the stationarity analysis method FSS10.

The verification experiments are presented as follows. A time series is shown in Fig. 4.2a. It was obtained using the uniform distributed PRNG included in Matlab. This signal is used as a reference to verify that the proposed tool is implemented properly. The size for the time series

realization $\{x_t\}_{t=1}^{n}$ is $n = 245,760$. This length fulfills all the requirements stated previously for the windows length. Figure 4.2b shows a pdf with uniform distribution. This plot shows the range between the maximum and minimum values in the pdf to provide a clearer appreciation of the values variations.

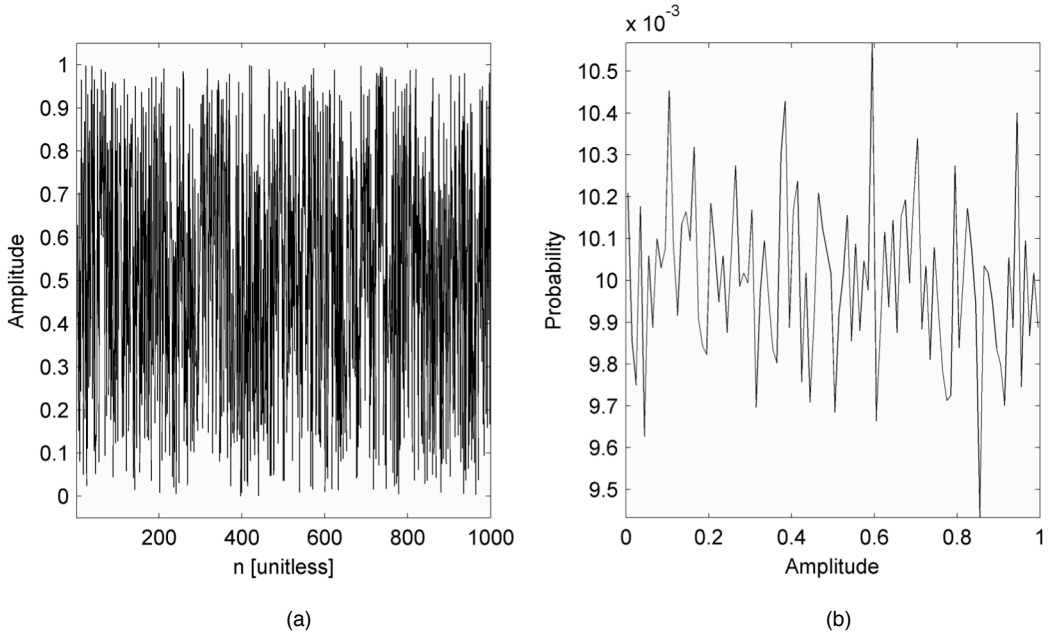


Fig. 4.2. (a) Time series with uniform distribution. This was obtained using the PRNG included in Matlab. Just the first 1000 samples are shown for clarity. **(b)** Normalized uniform pdf of the complete time series realization. The number of bins used is 100.

The analysis of the first ten statistical moments of the time series shown in Fig. 4.2a is presented in Fig. 4.3. The window length is $n_w = 2^{13}$. Figure 4.4 displays a plot that subtracts the minimum value of each moment series for better appreciation of the different moments variations. It is shown in Fig. 4.3a that the zetoth order moment has a value of one. This value pre-verifies that the constructed analysis tool is implemented correctly. The statistical moments presented in Fig. 4.3b appear to be constant. In figure 4.4, it is shown that the differences caused

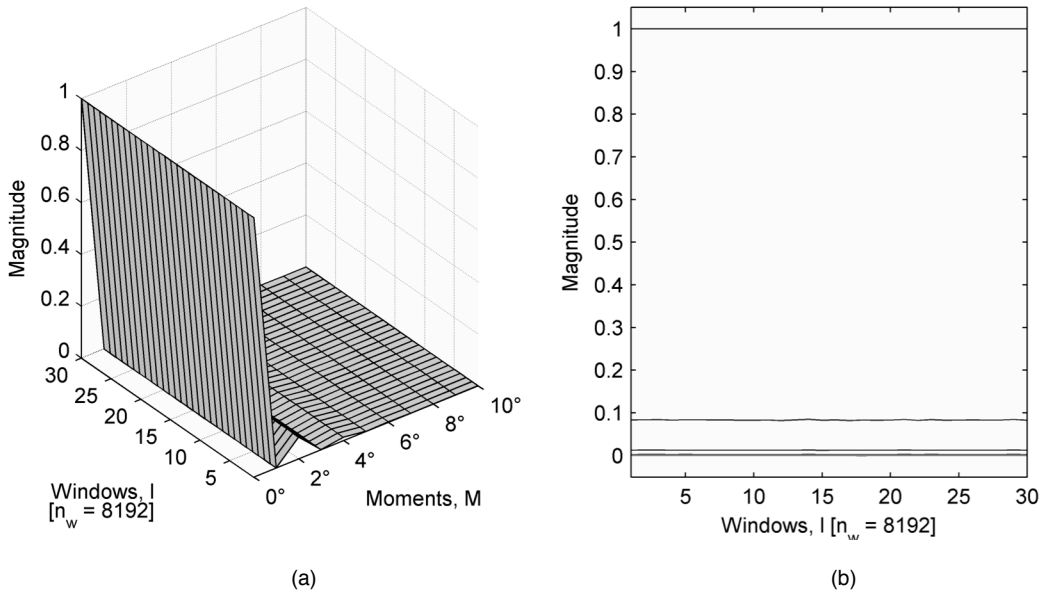


Fig. 4.3. **(a)** 3D representation of the first ten moments analysis of a time series with uniform distribution. **(b)** 2D representation of the first ten moments analysis of a time series with uniform distribution. The window length is $n_w = 2^{13}$ in both cases.

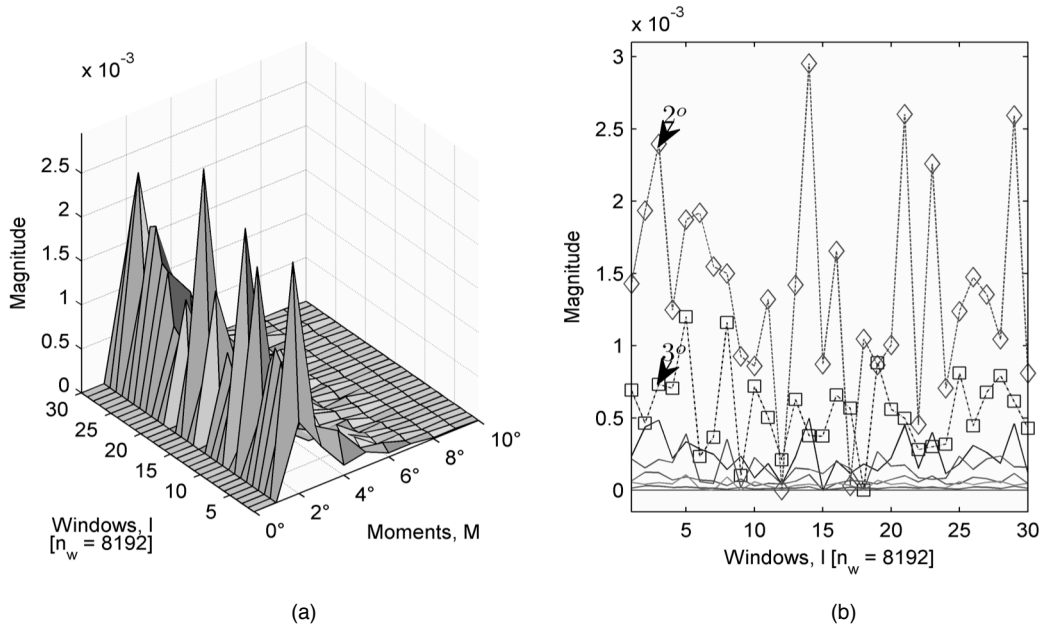


Fig. 4.4. **(a)** 3D difference representation of the first ten moments analysis of a time series with uniform distribution. **(b)** 2D difference representation of the first ten moments analysis of a time series with uniform distribution. The window length is $n_w = 2^{13}$ in both cases.

(i) a value displacement for all the statistical moments towards a common reference, being this zero and (ii) improved insight into the statistical moments that have more complex variations. It is concluded from Fig. 4.4 that the statistical moments with more complexity are the variance (2^{nd}) and the skewness (3^{rd}) that are identified by a diamond and square marker respectively.

Noting the advantages of presenting the moments in plots based on differences, Fig 4.5

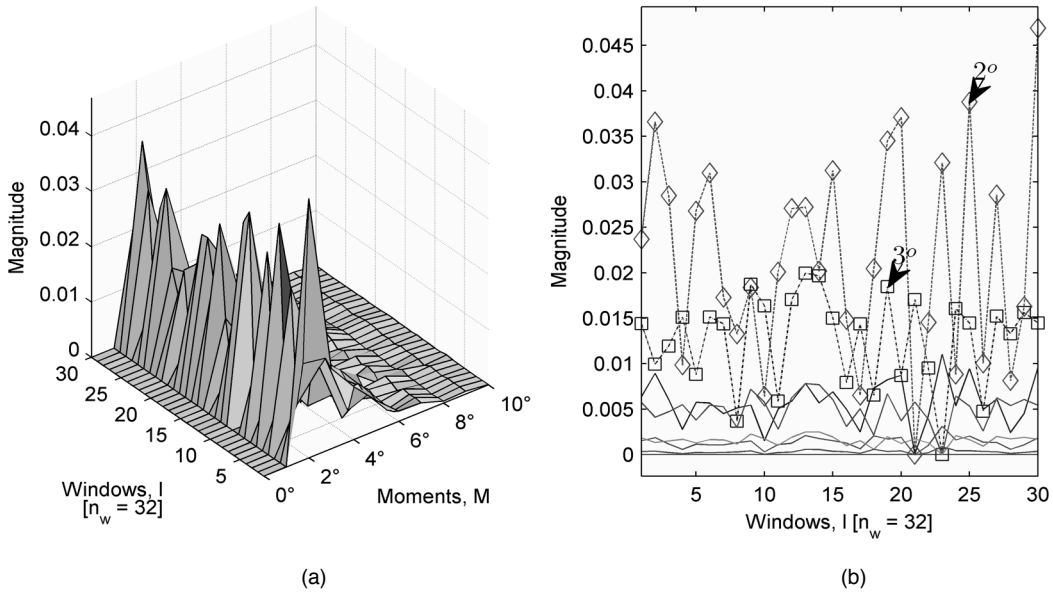


Fig. 4.5. (a) 3D difference representation of the first ten moments analysis of a time series with uniform distribution. **(b)** 2D difference representation of the first ten moments analysis of a time series with uniform distribution. The window length is $n_w = 32$ in both cases.

denotes the application of this technique used in the previous figure for a window length of $n_w = 2^5$. The variance (2^{nd}) and skewness (3^{rd}) continue varying in the same way as in Fig. 4.4.

It is seen that the variation of the 2^{nd} and 3^{rd} statistical moments is in the order of thousandths for a window length $n_w = 2^{13}$ as depicted in Fig. 4.4 and in the order of hundredths for a window length $n_w = 2^5$ as shown in Fig. 4.5. The first 10 statistical moments of interest are analyzed considering different windows sizes where their values are shown simultaneously in Fig. 4.6.

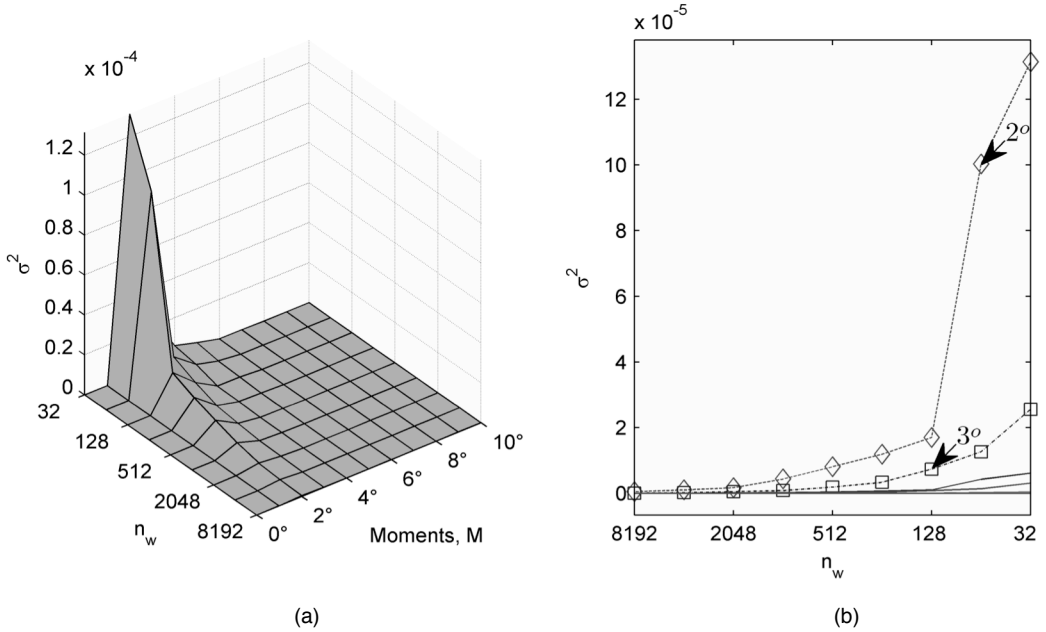


Fig. 4.6. (a) *Stationarity map* based on $\boldsymbol{\sigma^2}$ (variance) of a uniform distributed time series. **(b)** 2D representation of the *stationarity map*. The 2nd and 3rd moments are highlighted due to their variation. The window length goes from $n_w = 2^{13}$ to $n_w = 2^5$ in both cases.

The order of statistical moments are restricted to the tenth degree in this compact representation caused by overlooking higher order moments, which vanish to zero and thus provide no significance in this stationarity analysis FSS10. The variance σ^2 is used in order to get this compact representation of the degree of variation of a given moment series. This illustrates how spread out the values of a given statistical moment are over the successive windows analyzed. This measure of dispersion of the statistical moments is significant because it is directly connected with the moments time invariance, hence their stationarity. An extension of this analysis is can be addressed using cumulants. Figure 4.6 presents in a compact form the variance of the first ten moments with window lengths from $n_w = 2^{13}$ to $n_w = 2^5$. This figure allows analyzing the variance of different moments having different window sizes simultaneously. Due to this reason, the plot presented in Fig. 4.6a is referred as *stationarity map* in this thesis.

The significance of the stationarity map for this research is quite substantial given the fact that a robust method to determine the correct window size to analyze stationarity in a time series has not been reported in the literature so far. The stationarity map in Fig. 4.6 allows an easy and fast exploration of the moments variation. It is seen in this plot that the moments with higher variance are the 2nd and 3rd which coincides with the information extracted from Figs. 4.4 and 4.5. Figure 4.6 provides an overview of the statistical moments, considered by FSS10, to determine precisely (i) the windows size in which a time series is stationary, (ii) detect variance increments of the statistical moments calculated, and (iii) determine how extensive those increments are. Looking at these factors, Fig. 4.6b shows the biggest variance increment for the time series when a change occurs from window $n_w = 2^7$ to $n_w = 2^6$.

The *stationarity map* can provide conclusions in the analysis of a time series that are by far easier to interpret than (i) moments analysed by single window sizes as is the case of Figs 4.4 and 4.5 and (ii) direct analysis of the statistical moments variation for different windows sizes like Fig. 4.7. However, Fig. 4.7 verifies that the variance (2nd) and skewness (3rd), shown in Fig. 4.7a and Fig. 4.7b respectively, vary accordingly with the reduction in the window size. In other words, the fewer number of samples that are present in the window in which a given statistical moment is computed, there is a noticeable variation between successive windows.

The stationarity property in a signal depends entirely on the process from which the signal is measured. This relates to the physics of the process in the case of signals found in nature (*e.g.*, voice). In this sense, the stationarity map is helpful to identify the appropriate window length in which a given signal is stationary. This signal can be natural or synthetically generated by computers, as is the case of the cryptosystem presented here.

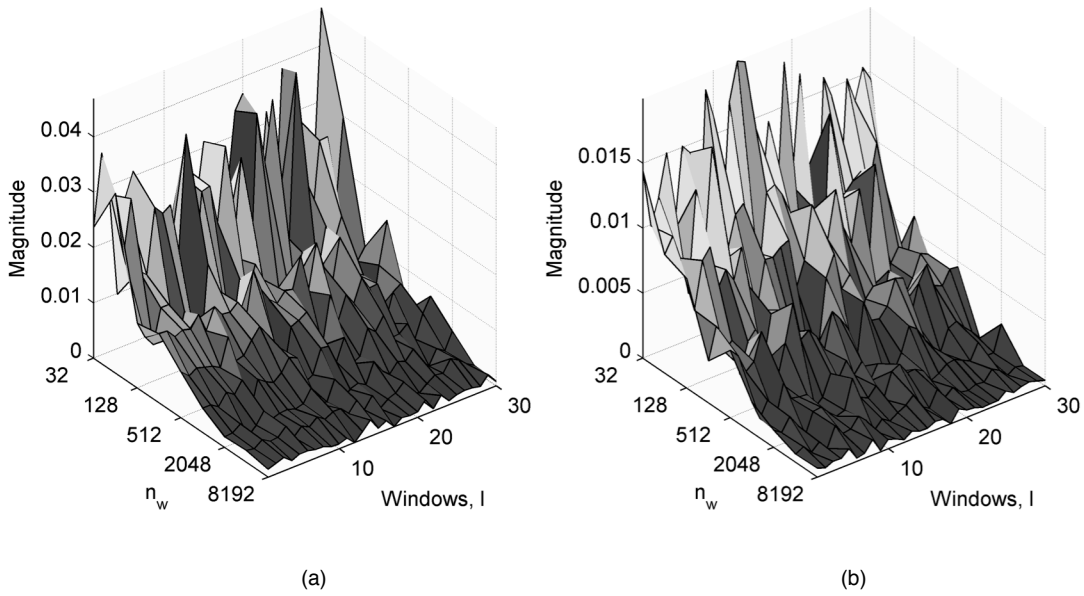


Fig. 4.7. (a) Variation of the 2nd moment of a uniform distributed time series for different windows sizes. **(b)** Variation of the 3rd moment of a uniform distributed time series for different windows sizes. The window length goes from $n_w = 2^{13}$ to $n_w = 2^5$ in both cases.

Considering that the maximum value located in the peak shown in Fig. 4.6 has a negligible value of approximately 1×10^{-4}, the following is concluded about the uniform distributed time series stationarity: (i) the time series under analysis is FSS10 stationary, (ii) the tool developed to test stationarity in time series has been verified to work properly, and (iii) the stationarity analysis tool developed has been proven to be valid, even when a small number of elements are present in a window (*i.e.*, less than a thousand). The analysis method FSS10 establishes a higher stationarity sense than just WSS because it is capable of examining higher order statistical moments.

The complete set of figures showing the variations of the statistical moments using different window sizes is included in Appendix A. In this subsection, a practical method has been created and verified that (i) properly defines the window length in which a signal is considered stationary

and (ii) concludes if a signal is FSS10 stationary.

4.2 Variance Fractal Dimension

This subsection describes a polyscale methodology that measures the complexity of a signal [Kins11b]. Polyscale analysis requires a measurement process in which the scale used is multiplied or divided by a constant factor at each stage. This allows access to the object properties for analysis based on the size of the scale used at a given stage. If the properties vary at different scales as a power law, then it can be determined that the object under analysis is a fractal, and a fractal dimension for those properties can be obtained. A fractal dimension is an indicator of the degree of complexity of an object, pattern, or in this case a sequence. Another method to assess whether a time series is chaotic includes *Lyapunov exponents* ([DeGu10], [Kins03], and [Kins11a]). Different fractal dimensions that are based on (i) the morphology (*e.g.*, self-similarity, Hausdorff, Minkowski, mass, gyration, and Lyapunov fractal dimensions), (ii) the entropy (*e.g.*, information, correlation, Rényi spectrum, and Mandelbrot singularity spectrum fractal dimensions), and (iii) the accurate analysis of time series (*e.g.*, variance and spectral fractal dimensions) are reported in the literature [Kins11a]. A unification of fractal dimensions is proposed by Kinsner [Kins08]. The research presented in this thesis focuses in the variance and spectral fractal dimensions because they can handle time series explicitly.

The calculation of a fractal dimension in terms of variance known as *variance fractal dimension* (VFD) ([Kins11a], [KiGr08], [KiGr10], [Kins07c], and [KCCP06]) is used as a tool to determine the complexity of signals produced by natural phenomena or synthetized by computers. This subsection verifies the implementation of the VFD algorithm by relying on a known process, as is the case of *Gaussian white noise* (GWN) and uniform distributed noise.

This tool is used further in the next chapter for testing if the cryptosystem is indeed fractal.

A time series can be analysed directly in time by computing the spread of the increments in the signal amplitude (*i.e.*, through its polyscale variance denoted as σ^2. The variance fractal dimension can be computed in real-time [Kins11a]. An important characteristic of the VFD is that it does not require a window in the Fourier sense, and therefore does not introduce corresponding artifacts [Kins11a].

The variance fractal dimension, D_σ, is determined by the Hurst exponent, H. The variance, σ^2, of the amplitude increments of a signal $B(t)$ over a time increment Δt is related to the time increment according to the following power law [Kins11a]

$$\mathrm{var}\left[B(t_2)-B(t_1)\right] \sim |t_2-t_1|^{2H} \tag{4.3}$$

where var denotes variance, and the symbol $\sim$ reads "is proportional to."

For $\Delta t = |t_2 - t_1|$ and $(\Delta B)_{\Delta t} = B(t_2) - B(t_1)$ the exponent H can be calculated from a $\log - \log$ plot by Shannon [Shan49]

$$H = \lim_{\Delta t \to 0} \frac{1}{2}\left(\frac{\log_b\left[\mathrm{var}(\Delta B)_{\Delta t}\right]}{\log_b \Delta t} \right) \tag{4.4}$$

in the analysis performed here, the base b is 2. The embedding Euclidean dimension E (*i.e.*, the number of independent variables in the signal under analysis), the variance dimension can be

computed from

$$D_\sigma = E + 1 - H,\ 1 \le D_\sigma \le 2 \text{ and } 0 \le H \le 1 \tag{4.5}$$

The implementation of the technique to calculate the VFD in a digital signal consists of the following steps [Kins11a]: first, a sample space of N_T points from the signal is chosen. The range of sizes of Δt at which the spread of ΔB should be computed is obtained by $\Delta t_{K\max} = n_{K\max}$, where $\delta t \le T$. The time interval δt should not exceed the total time T over within the sample space. The parameters for the loop computation of the variance are prepared as follows:

$$K_{\max} = \text{int}\left(\frac{\log_b N_T}{\log_b} \right) \tag{4.6}$$

where $b = 2$, in this case, is the base to form a b-adic sequence for time intervals n_k; $K_{\text{buf}} = \left\lceil \log_b(8{,}192) \big/ \log b \right\rceil$, where $N_T = 8{,}192$ (desirable to be greater than 30 for statistical significance) represents the number of divisions in the first computation in the loop; $K_{\text{hi}} = K_{\max} - K_{\text{buf}}$; and $K_{\text{low}} \ge 1$. The main loop to obtain the VFD performs k cycles from K_{hi} to $k = 1$ in which the number of samples is $n_k = b^k$. The number of windows in the signal is represented as $N_k = \text{int}\left(N_T \big/ n_k \right)$, and the variance for each stage is then

$$\mathrm{var}(\Delta B)_k = \left[\frac{1}{N_k - 1}\right]\left[\sum_{j=1}^{N_k}(\Delta B)_{jk}^2 - \frac{1}{N_k}\left[\sum_{j=1}^{N_k}(\Delta B)_j\right]^2\right] \tag{4.7}$$

The amplitude increment is given by

$$(\Delta B)_j = B(jn_k) - B((j-1)n_k) \text{ for } j = 1, \ldots, N_k \tag{4.8}$$

The log values $X_k = \log[n_k]$ and $Y_k = \log[\mathrm{var}(\Delta B)_k]$ are stored for the $\log - \log$ plot and the least-squares fit to obtain the slope s of the line is obtained using

$$s = \frac{K\sum_{i=1}^{K} X_i Y_i - \sum_{i=1}^{K} X_i \sum_{i=1}^{K} Y_i}{K\sum_{i=1}^{K} X_i^2 - \left(\sum_{i=1}^{K} X_i^2\right)^2} \tag{4.9}$$

The Hurst exponent is computed by $H = \left(1/2\right)s$, and the VFD is obtained by applying (4.5).

For a non-stationary sequence, this process is repeated on successive windows (either non-overlapping or overlapping) to obtain a *VFD trajectory* (VFDT) [Kins11a]. If the VFDT is constant then the sequence is a monofractal in time. Also, if the VFDT has segments with different slopes, the sequence is multifractal in time.

After presenting the method to obtain the VFDT known signal, GWN and uniform distributed noise, are used to verify its implementation. Figure 4.8 shows the log values X_k and

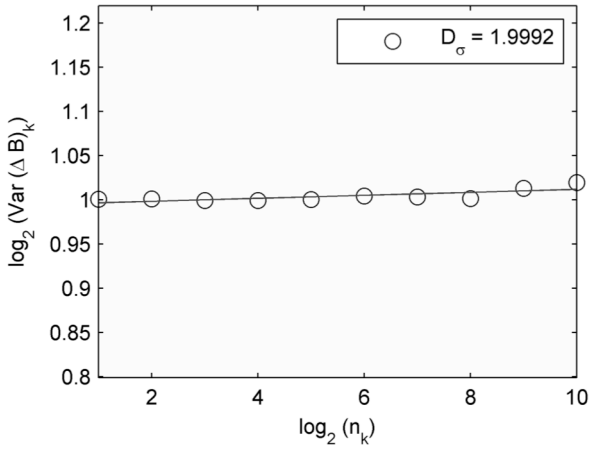


Fig. 4.8. Gaussian white noise VFD. This is based on a sequence realization of 10 million. The last ten binary orders of magnitude in the computation are displayed. As expected the variance is 1, and the variance fractal dimension is almost 2.

Y_k and the slope fitting applying the procedure just described in the previous equation to a realization of a *Gaussian white noise* (GWN) signal by analyzing 10 million samples. The variance fractal dimension for the GWN signal is $D_\sigma \cong 2$, which is expected for a space-filling function. The parameters used for the loop computation of the VFD are $K_{\max} = 23$, $K_{\text{buf}} = 13$, $K_{\text{hi}} = 10$, and $K_{\text{low}} = 1$. As a general rule, the first possible ten points in the log-log plot are not considered in any of the VFD computed due to some artifacts that could arise. Just the strongest part where the slope is preserved is computed and displayed.

The realization of a uniformly distributed random sequence in the interval $[0,1]$, where the same conditions are considered, is displayed if Fig. 4.9. The variance, in this case, is quite small

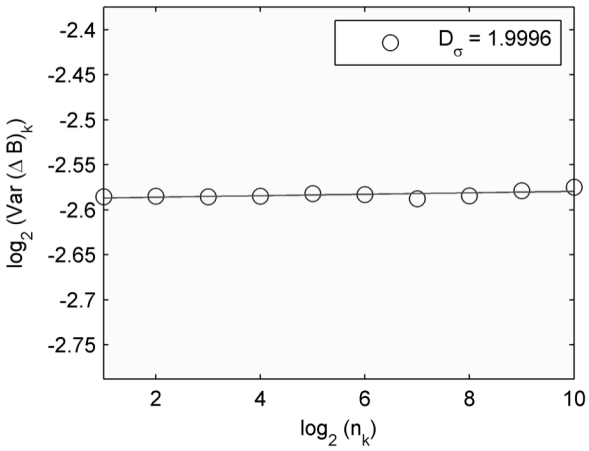


Fig. 4.9. Uniformly distributed on the interval $[0,1]$ pseudorandom numbers VFD. This is based on a sequence realization of 10 million. The last ten binary orders of magnitude in the computation are displayed.

as compared with the GWN analysis presented previously. The negative values displayed in the y axis in the figure are due to the logarithmic base $b = 2$ representations of the variance values. It should not be confused with an attempt to represent negative variances, which would be against to the mathematical formulations and concepts of probability.

4.3 Spectral Fractal Dimension

Power spectrum analysis methods for time series by decomposing into a sum of sinusoidal components, as is the case of the Fourier transform, are robust against changes in amplitude. The spectrum measures the *probable amplitude* of the oscillations in a series as a function of the frequency. Therefore, attention is diverted away from the search for unknown periodicities in a time series toward the usually more informative study of the relative amplitudes at *all* frequencies [Bloo06]. The most critical drawback is that by just taking the Fourier transform over a signal is not a guarantee to determine its properties totally. This limitation is caused by the

time information loss when applying the Fourier transform. This transformation provides a barrier to linking a given event with time. Stationary signals do not have this problem. In the case of signals that have transients, or time varying signals, it is important to determine at what point in time the signal exhibited a particular behaviour [Gröc01].

The power spectrum alone provides a qualitative sense on how complex a signal is, however it does not convey a quantitative measure of the signal complexity. Nevertheless, the *spectral fractal dimension* (SFD) D_β, which is based on the *power spectrum density* (PSD) analysis, is capable of determining the degree of complexity of any given signal. This analysis technique can be used in globally or locally stationary signals. For the last case, this type of analysis considers small portions of the signal, or windows, in which it is stationary. This analysis tool is also helpful for signals that are composed of a small number of samples and is also suitable to deal explicitly with time series [Kins11].

A time series can be transformed into its power spectrum density $P(f,T)$, where f is the frequency and T is the time interval (length) during which the time series is stationary, using spectral analysis techniques (*i.e.*, the *Fast Fourier Transform* (FFT) and *discrete cosine transform* (DCT)) [Kins11a]. Time-scale analysis such as a wavelet transform may also be used ([Daub92] and [Kisn91]).

The useful information that the SFD can provide are: (i) revealing the richness of the different frequencies representing the underlying process behind the time series, (ii) determining if the underlying process is *periodic* (non-chaotic), given a power spectrum with equally spaced sharp lines (harmonics), and (iii) determining if a broadband power spectrum might have originated from chaos by presenting substantial power at low frequencies (this is not a guarantee

for the process to be sensitive to initial conditions) [Kins11a].

When a signal power spectrum density satisfies the following power law then the time series is considered as a self-affine fractal.

$$P(f,T) \sim \frac{1}{f^{\beta}} \tag{4.10}$$

For a self-affine time series with a single independent variable $(E=1)$ ([Turc97] and [Kins11a]), the *spectral fractal dimension* (FSD) is defined as

$$D_{\beta} = E + \frac{3-\beta}{2} \tag{4.11}$$

Determining the spectral fractal dimension of a time series with uniform distribution generated by the Matlab PRNG serves as a basis to verify the capabilities of the SFD as an analytical tool. The time series generated in Fig. 4.2a is used to verify the SFD, where a sequence with length $n=128$ for the time series realization $\{x_t\}_{t=1}^{n}$ is used. This value selection is consistent with the conclusion obtained in section 4A, where the Matlab PRNG is verified to be strongly stationary for a minimum window length $n_w = 2^7$ (observe Fig.4.6). The value of the power spectrum exponent for this white noise time series is known to be $\beta = 0$ [Kins11a]. The power spectrum density of the referred time series is shown in Fig. 4.10.

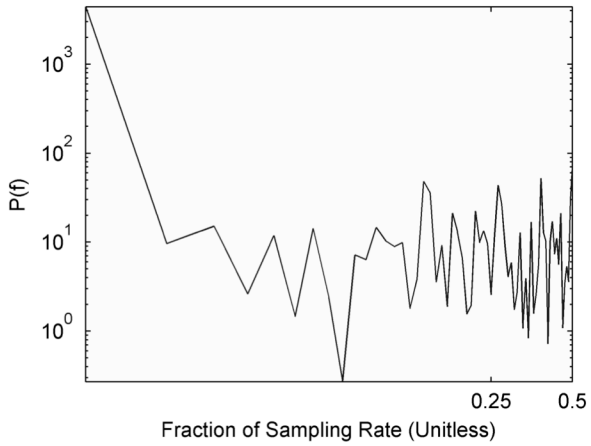


Fig. 4.10. Matlab PRNG power spectrum density displayed in a log – log plot.

The power spectrum density displayed in Fig. 4.10 is used to determine the power spectrum exponent β by fitting a straight line. In Fig. 4.10 the low frequencies appear to be stretched while the high frequencies compressed. This behaviour is known as *Spanish moss* effect. The presence of Spanish moss effect in the PSD blocks the perception of the true envelope.

The problem that arises in Fig. 4.10 is that it appears to be not significant to fit a straight line in a plot that does not have equal spacing in its support. Before fitting a straight line, a special procedure to reduce the Spanish moss effect is required. One way is to divide the support axis in equally divided bins and average the spectrum within each bin [Kins11a]. However, a more scientific approach would be to find the appropriate bin size that could contain the corresponding part of the spectrum to be averaged. It is important to underline that this problem is not trivial; therefore, in order to have a reliable analytical tool that is capable of providing significant results, this problem has to be addressed.

An intuitive approach to the division of a time series power spectrum density into appropriate bins is to create an analogy using the process of making music. The piano helps to explore how

the frequencies behave in the power spectrum. A piano is truly a precision machine, with 88 keys, typically with 230 strings, and approximately 10,000 individual parts. An important component to creating sound that is external to the piano is the air around it. The air allows the piano to carry the sound waves, with a certain pitch produced by the strings, to travel through to the listener [Gior10].

Once the combinations of tones produced by a piano reaches the listener her/his auditory system processes the sound. This is done by the detection of sound that the small oscillatory forces exert on the eardrum by the pressure variations associated with a sound wave. The ear is extremely sensitive and can detect pressure oscillations from a sound wave of less than one billionth $\left(10^{-9}\right)$ of normal atmospheric pressure. The eardrum acts on the cochlea that has a spiral shaped structure. Here, the ear generates an electrical signal that is sent to the brain via the auditory nerve. Different parts in the spiral shape of the cochlea respond or detect different frequencies. The ear of a young adult can discern frequencies in the range of about $20-20{,}000$ Hz, although the upper limit decreases significantly with age [Gior10].

Figure 4.11 shows a piano octave. The term octave is coined from the fact that in a given scale it requires eight notes for one musical pitch to reach another pitch with half or double its frequency moving towards the low or high notes respectively [Coop81]. The notes and the scale(s) that can be derived from them are familiar from western music. The white keys shown in Fig. 4.11 can be used to play the scale $do-re-mi-fa-sol-la-si-do$. This scale starts and ends with the "same" note, in this particular case called do, as is the case for all the 12 scales. This scale can be written as [Gior10]

$$C-D-E-F-G-A-B-C \tag{4.12}$$

The notes in Fig. 4.11 repeat every 12 notes in either direction right or left in a piano, even

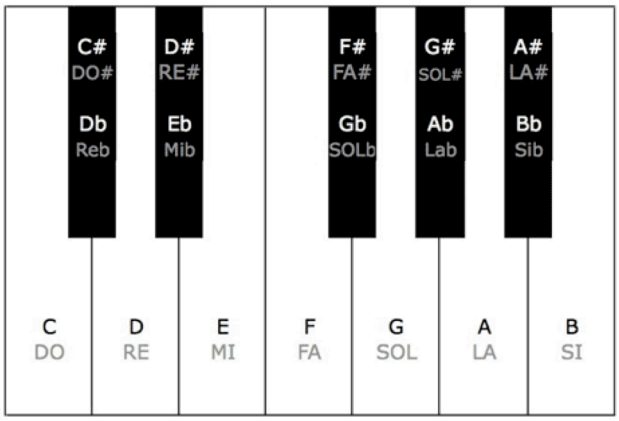


Fig. 4.11. Piano octave.

though their fundamental frequencies and hence their pitches are different. This notion of sameness is purely based on individual perception. Nevertheless, it can serve as an indicator about the behaviour of frequencies in the power spectrum.

Pressing the white keys of a piano simultaneously is an experiment that allows exploring how frequencies are related in the power spectrum. In order to this, the synthetized version of this sound is obtained using

$$P(key) = P(key_a)\left(\sqrt[12]{2}\right)^{(key-key_a)}, \text{ for } key = 1,\ 2,\ ...,\ 88 \text{ and } key_a = 49 \tag{4.13}$$

where $P(key)$ refers to the pitch, or frequency in Hertz, that a particular piano key has in reference to another pitch $P(key_a)$ (usually 440 Hz corresponding to A4, the 49th key from left

to right in the piano keyboard). A slight modification to equation 4.13 is to use a value for $P(key_a) = 431$ to yield a *scientific pitch* ([Youn39] and [Gran11]) which doubles the frequency of a given note at each octave moving towards the high tones in a piano keyboard in a pattern that resembles powers of two for the C notes. The following equation shows this pattern explicitly

$$P(key) = 431\left(\sqrt[12]{2}\right)^{(key-49)}, \text{ for } key = 1,\ 2,\ ...,\ 88 \tag{4.14}$$

An important constant that both equations 4.13 and 4.14 include is $\sqrt[12]{2}$. The significance of this is that it allows having *equal temperament* between adjacent notes. Equal temperament is a system of tuning that allows every pair of adjacent notes to have an identical frequency ratio [Murr04]. An illustration of this is the frequencies of an octave divided into a series of equal steps in the logarithmic scale (usually 12 for western music or 24 for Arabic music). The constant $\sqrt[12]{2}$ then becomes the ratio between two adjacent notes [Helm54].

All white key frequencies, from the C scale, in the eight piano octaves are generated into independent signals. After their generation, they are synthesized a single signal containing all the key frequencies using

$$P_C = \sum_{key=1}^{88} P(key) \tag{4.15}$$

where P_C denotes the tones pitch contained in the C scale and key must correspond to a white

key. The black keys that do not correspond to the C scale are excluded in the practical implementation of P_C.

The power spectrum density corresponding to the P_C signal is shown in Fig. 4.12. Equally spaced segments in the logarithmic horizontal axis support the octaves.

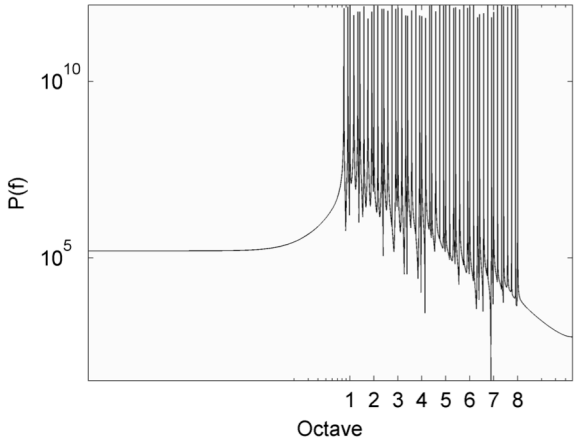


Fig. 4.12. Power spectrum density of the C scale in a piano. The presence of the *Spanish moss* effect is shown.

A theoretical piano tuned with scientific pitch by equation (4.13) could reach a frequency as small as $P(key)=1$ Hz by dividing its frequency in halves in the logarithmic scale at each theoretical octave below its octave one. Under this observation, it can be concluded by averaging the frequencies in the PSD support of a given octave can diminish or eradicate completely the effects of the *Spanish moss* and equally spacing occurs as seen in Fig. 4.13. In this figure it is shown that a diagonal line, denoted by the circles, emerges resembling a power law for all eight piano octaves. A superimposed line on the piano octaves is also shown which the diamonds denote. This line was obtained by a weighted least squares fitting process using the equations [Moon99]

$$\mathbf{c} = \left(\mathbf{A}^H \mathbf{W} \mathbf{A}\right)^{-1} \mathbf{A}^H \mathbf{W} \mathbf{y} \tag{4.16}$$

$$\hat{\mathbf{y}} = \mathbf{A}\mathbf{c} \tag{4.17}$$

with the modification to operate in logarithm base 10 as follows

$$\mathbf{c} = \left(\mathbf{A}^H \mathbf{W} \mathbf{A}\right)^{-1} \mathbf{A}^H \mathbf{W} \log_{10} \mathbf{y} \tag{4.18}$$

where $\mathbf{c}$ contains the coefficients which minimize the weighted square error, the term $\mathbf{A}^H\mathbf{A}$ known as the *Grammian operator* used in projections includes a weighted matrix $\mathbf{W}$, $\mathbf{y}$ is the vector containing the vertical values for each averaged frequency and $\hat{\mathbf{y}}$ is the projection vector that yields the straight line seen in Fig. 4.13 denoted by diamonds. The matrix $\mathbf{W}$ weighs the

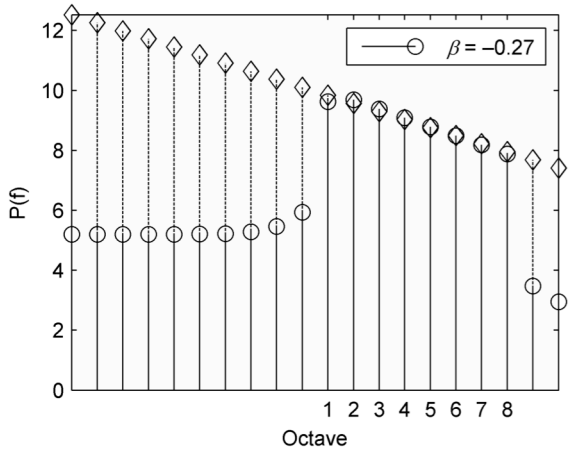


Fig. 4.13. Power spectrum density of the C scale in a piano. The *Spanish moss* effect is eliminated by averaging.

data points that are confident, which, in this case are the averaged frequencies at a given piano scale from one to eight.

Once the straight line denoted by diamonds has been fitted it is possible to determine its slope by using the equation

$$m = \frac{\Delta y}{\Delta x} = \frac{y_2 - y_1}{x_2 - x_1} \tag{4.19}$$

which is equivalent to the value for the power spectrum exponent β. It is shown that the power spectrum exponent $\beta \sim 12 \log_{10}\left(\sqrt[12]{2}\right)$. The inclusion of the value 12 is due to the number of equal steps in the logarithmic scale to cover one single octave in the western equal temperament tuning system. After this immersion into music to exemplify how frequencies behave in the power spectrum and how the *Spanish moss* effect could be reduced, a robust analysis tool has been verified.

Figure 4.14 presents the results of applying this analysis tool to a signal obtained using the Matlab PRNG. The first step is to eliminate the *Spanish moss*. After this, the calculation of its power spectrum exponent is performed.

It is shown that the power spectrum exponent $\beta \sim 0$ is expected for a white noise time series. The first window is discerned by the weighted least squares fitting process using (4.16-18). The diamonds in Fig. 4.14 denote the fitting line. This experiment with the Matlab PRNG also serves to verify that the tool designed to further test the cryptosystem power spectrum exponent provides successful results across a known time series, as is the case of white noise.

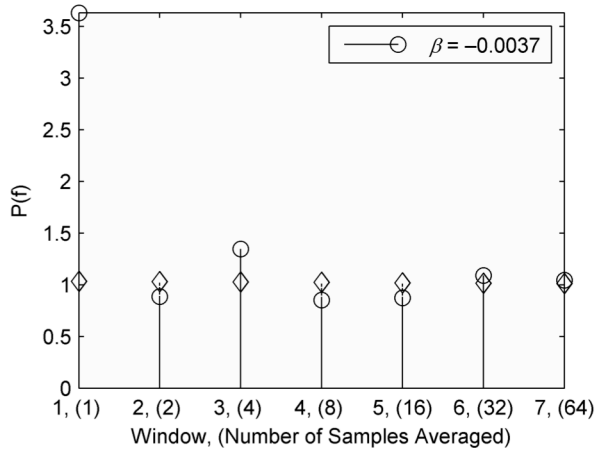


Fig. 4.14. Power spectrum density of the Matlab PRNG. The *Spanish moss* effect is eliminated by averaging.

4.4 Surrogate Data

Before applying nonlinear analysis techniques to a time series, it is necessary to first ask if the use of such advanced techniques is justified *by the data*. Different processes seem very unlikely to be linear, but their possible nonlinear nature might not be evident in their dynamics. Surrogate data addresses this question.

Surrogate data can be defined as the generation of alternative sequences, which are derived from a time series belonging to natural or abstract phenomena. The different methods to generate surrogate data aim to (i) preserve the time series original statistical properties and (ii) provide a totally different new sequence. These sequences can then be used to assess nonlinear features in the time series from which they are derived.

Nonlinear approaches to analyze a time series are motivated by these reasons: (i) intrinsic to the signal itself because certain structures in the signal are not possible to analyze relying on linear methods and (ii) additional knowledge that could be gained about the phenomenon under

analysis in a system that is known to have nonlinear components (*i.e.*, the brain having neurons which are nonlinear by nature) [ShSh11].

However, a system having nonlinear components does not imply its nonlinearity to be reflected in a signal measured from the system. Therefore, it is not known if it is practical to go beyond a linear approximation when analyzing a signal. As a consequence, application of nonlinear time series methods needs justification by establishing nonlinearity in the time series [ShSh11].

The construction of a surrogate time series considers the inclusion of the same properties as the original time series, but it is intended to be as random as possible. Once the surrogates have been created it is then when additional structures present in the original data, but not in the surrogates, can be analyzed. A simple method to generate a surrogate time series can be obtained by randomly shuffling the measured data (*i.e.*, random temporal order) [ShSh11]. This method is the simplest to generate surrogate data [Dola01] and can be helpful to hide a given chaotic attractor describing a dynamical system. Even though Theiler *et al*. showed that shuffled surrogates are unsuitable for nonlinearity testing [Dola01], this type of surrogates is capable of destroying the correlation from one point to another in a given signal. Constrained realizations can be obtained by creating permutations without replacement. The surrogates are also constrained to take on exactly the same values as the data, but reorganized in random temporal order [ShSh11]. Random shuffled surrogates are suitable for cryptographic purposes because they allow encryption and decryption processes to be performed without data loss. Surrogates that modify the data amplitude and are prone to rounding errors (*i.e.*, Fourier based) are not useful for cryptographic purposes because the PT might become altered by the data loss. Using

surrogates provides new data that is at some degree comparable, but not equal to the data from which they are derived. In this sense, surrogate data provides alternative data that cannot be related with the data source directly. This surrogates type is beneficial for chaos-based cryptography because the chaotic process used to encipher/decipher a secret cannot be monitored and thus broken. Random shuffled surrogates are helpful to conceal the cryptosystem chaotic attractor leading to a significant enhancement of the cryptosystem security. Another characteristic that makes random shuffled surrogates very attractive for chaos-based cryptography is that all statistical moments are not modified. This provides an infinite number of possible data sets with exactly the same statistical characteristics.

Now let us verify how the presented ideas can conceal a given chaotic attractor. This is exemplified using the Hénon attractor, which is described by [KaSh04]:

$$x[n] = a - \left(x[n-1]\right)^2 + b\left(y[n-1]\right) \tag{4.20}$$

$$y[n] = x[n-1] \tag{4.21}$$

where the parameters, selected to have a chaotic regime, are $a = 1.4$ and $b = 0.3$ (for $b = 0.3$, chaotic behaviour is observed by varying the parameter a in the range from $a = 1.05$ to $a = 1.43$ approximately). Figure 4.15a shows part of the realization of a time series 250,000 samples long. After randomly shuffling this time series, by visual inspection it is seen in Fig. 4.15b that the surrogate obtained resembles the original time series. However, the correlation from one sample to another seems to be gone. In order to explore the extent of this effect, it is

necessary to compare the Hénon chaotic attractor in its canonical shape and after obtaining a randomly shuffled surrogate. To achieve this, it is necessary to reconstruct the *phase space* from the time series displayed in Figs. 4.15a and 4.15b by the method of delays (or related

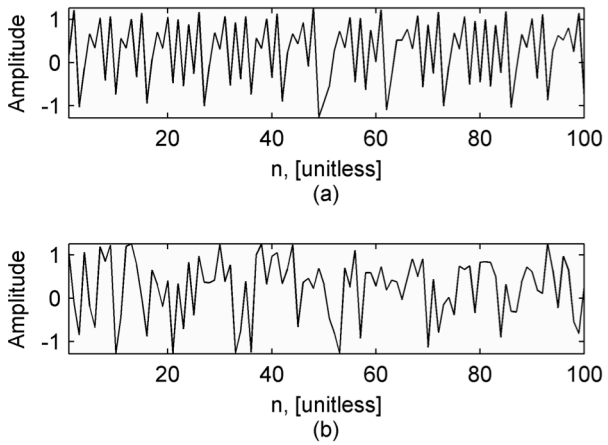


Fig. 4.15. (a) Time series obtained from the Hénon attractor. Just the first 100 samples are shown. **(b)** Surrogate data obtained through randomly shuffling the time series displayed in (a).

reconstructions) [KaSh04].

A *delay reconstruction* in m dimensions is formed using [KaSh04]

$$\mathbf{S}_n = \left(S_{n-(m-1)},\, S_{n-(m-2)},\, \ldots,\, S_n \right) \tag{4.22}$$

where $\mathbf{S}_n$ represent the vectors required to perform the phase space reconstruction. Under quite general circumstances, the attractor formed by $\mathbf{S}_n$ is equivalent to the attractor in the unknown space in which the original system is living [KaSh04].

The Hénon attractor phase space shown in Fig. 4.16 is reconstructed using (4.22) and $m = 2$.

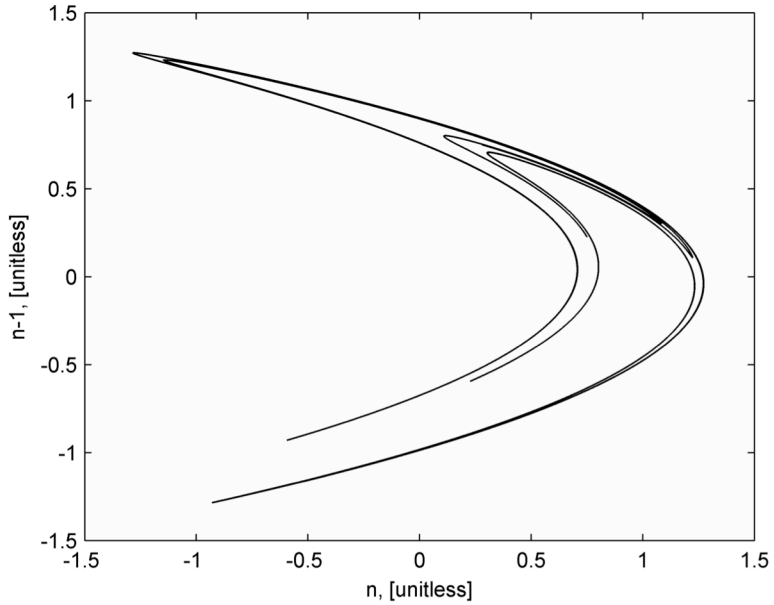


Fig. 4.16. Hénon attractor. A time series 250,000 samples long is used.

After performing a surrogate and attempting to reconstruct the phase space with the same method of delays no trace of the chaotic attractor is left, as shown in Fig. 4.17. The achievement obtained

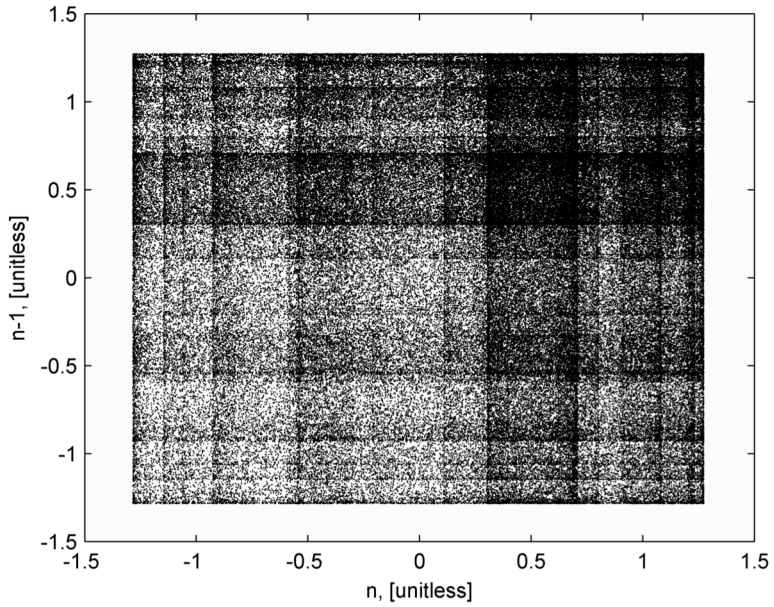


Fig. 4.17. Hénon attractor after obtaining a surrogate randomly shuffled. A time series 250,000 samples long is used.

through the use of surrogate data to conceal the Hénon attractor to conceal is significant. Applying the same method to the *multi-needle attractor* allows the cryptosystem to be stealthy and increases its security level. This is presented in the following chapter section four.

4.5 Summary

Four different signal analysis methods have been verified in this chapter; (i) the stationarity map, based on the FSS10 method, is capable of providing the minimal window size in which a signal is considered strongly stationary; (ii) the VFDT and, (iii) the SFD measure the complexity of a signal; while the use of (iv) surrogate data has revealed the capability to conceal a chaotic attractor. The first three methods provide a basis to produce quantitative measures when testing the cryptosystem properties and the latter enhances the security degree of the cryptosystem. These methods are also used in the next chapter for the same purposes.

CHAPTER V

ENCRYPTION SEQUENCES TESTING

Utilizing the methods developed and verified in the previous chapter it is then possible to analyze the designed cryptosystem by: (i) testing its stationarity properties, (ii) measuring its complexity, and (iii) demonstrating that its chaotic attractor can acquire stealth properties through the use of surrogate data. The implications obtained from the outcomes in this chapter are extremely important to measure quantitatively the degree of security offered by the cryptosystem. It is desired that this cryptosystem could count with strong stationarity properties, that the encryption sequence exhibits highly complex behaviour, and that the process that the cryptosystem uses to secure data is stealthy.

The stationarity map, based on the FSS10 method, is used to determine precisely the window in which the cryptosystem is strongly stationary, the VFD and SFD measure the cryptosystem complexity and compare it with the characteristics of white noise, and finally the method based on surrogates provides stealth capabilities to the cryptosystem and demonstrate that its chaotic attractor cannot be reconstructed by exposing state variables, attempting to build a model of the system, or finding synchronization patterns [YaWC97]. This ensures that the secrets, data, or information protected by means of this cryptosystem are secure and not easily breakable.

5.1 Stationarity FSS10

Testing the cryptosystem for stationarity requirements higher than those of WSS by using the FSS10 method allows determining if the cryptosystem can be considered secure from at least the point of view of time invariance. Figure 5.1 presents in a compact form the variance of the first

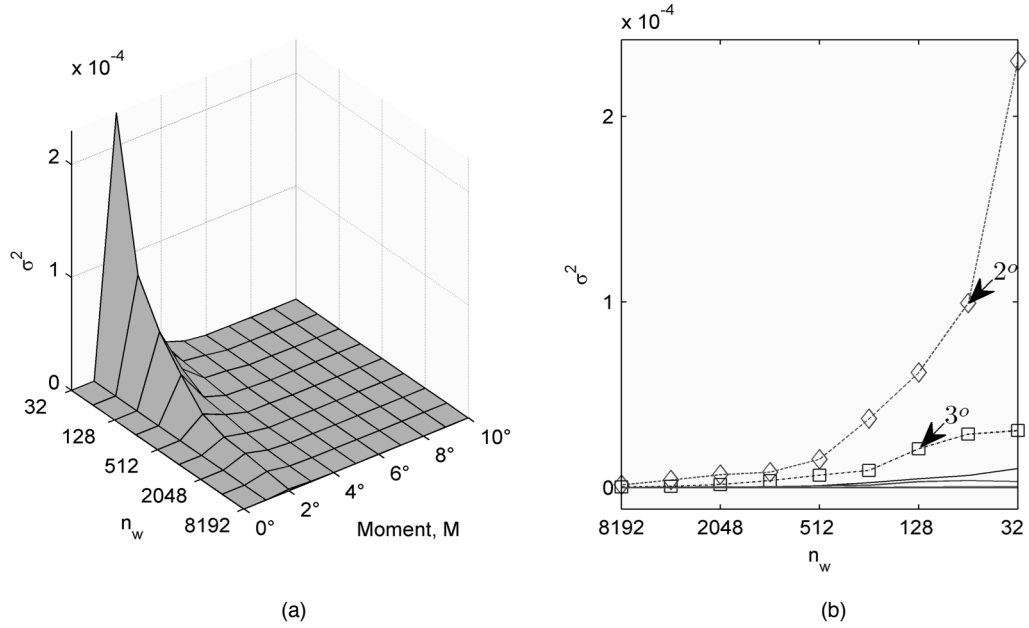


Fig. 5.1. **(a)** *Stationarity map* based on σ (standard deviation) of a time series produced by the cryptosystem. **(b)** 2D representation of the *stationarity map*. The 2nd and 3rd moments are highlighted due to their variation. The window length varies from $n_w = 2^{13}$ to $n_w = 2^5$ in both cases.

ten moments for a realization of an encryption sequence with size $n = 245{,}760$. This matches the size used in the previous chapter when verifying the FSS10 method. The same applies for the window lengths that vary from $n_w = 2^{13}$ to $n_w = 2^5$. From the stationarity map seen in figure 5.1 it is shown that the stationarity properties of the cryptosystem hold for a window size small as $n_w = 2^9$. This measurement is achieved using the robust FSS10 method verified in the previous chapter to determine the correct window size to analyze if a time series is FSS10 stationary.

Figure 5.2 shows the moments with higher variance from the time series yielded by the cryptosystem. This is a conclusion obtained from Fig 5.1 where the 2nd and 3rd moments are the more prominent values. In both figures 5.1 and 5.2, the values are slightly greater than the values obtained when the stationarity testing FSS10 is verified with the realization of a uniform time series. From this it is concluded that the cryptosystem resembles a random number generator

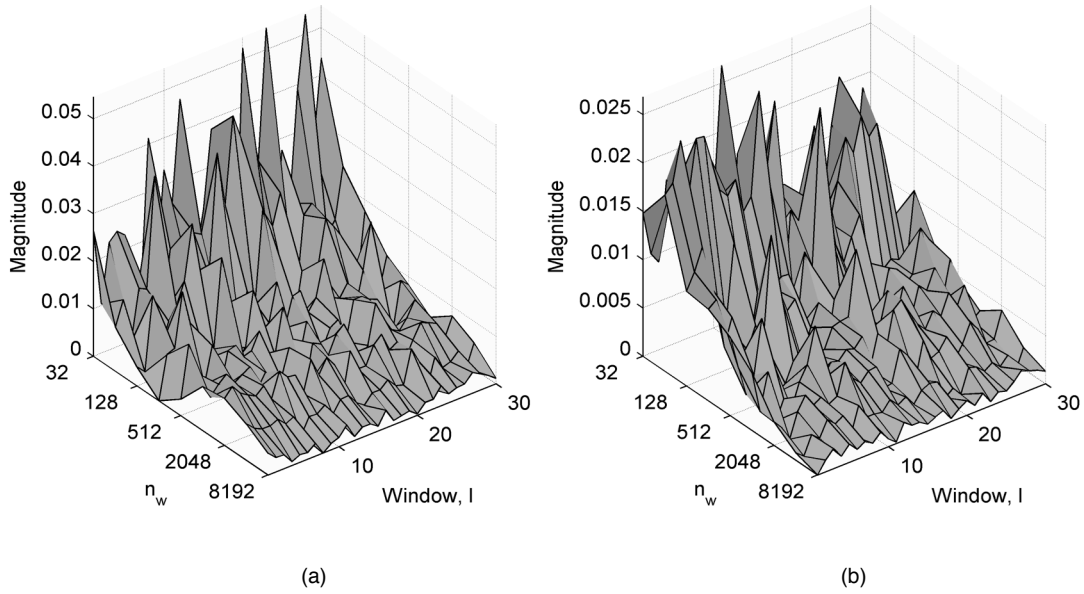


Fig. 5.2. **(a)** Variation of the 2nd moment of a time series produced by the cryptosystem with different windows sizes. **(b)** Variation of the 3rd moment of time series produced by the cryptosystem with different windows sizes. The window length varies from $n_w = 2^{13}$ to $n_w = 2^5$ in both cases.

with uniform distribution where stationarity holds for a window size greater than $n_w = 2^9$ as previously shown in Fig. 5.1. This directly relates with positive implications for the cryptosystem security. It is important to stress that the stationarity testing FSS10 performed here considers stricter constraints than those of WSS. This observation implies that the cryptosystem fulfills the requirements of WSS as well.

The stationarity map is an outstanding method to reveal precisely the window size in which a signal is FSS10 stationary. The significance of the stationarity map in this research is quite substantial, given the fact that a robust method to determine the correct window size to analyze stationarity in a time series has not been reported thus far in the literature. An objective appreciation to determine if a time series is stationary has been described through the development of the stationarity map. This method allows detecting variations in the moments values, and the extensiveness of these variations once the minimum window size in which it is

stationary has been found. By using the stationarity map, it is demonstrated that the cryptosystem resembles a RNG with uniform distribution where FSS10 holds for a window size greater than $n_w = 2^8$ samples.

5.2 Variance Fractal Dimension

This subsection uses the polyscale methodology, VFDT, described in the previous chapter in order to obtain a measure of the complexity of the cryptographic sequences both unmixed and chaotically mixed [Kins11b]. Figure 5.3 depicts the VFD of an unmixed realization of the

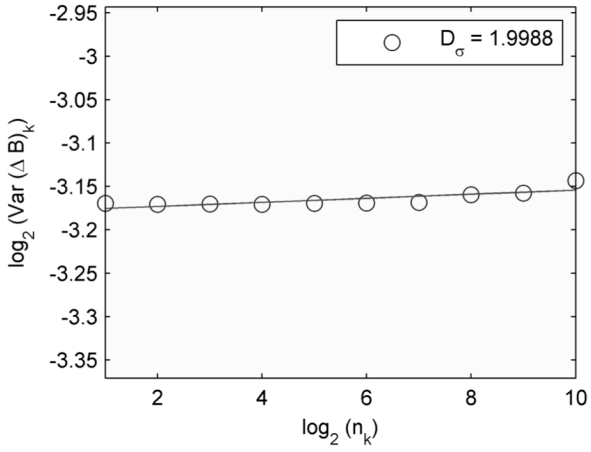


Fig. 5.3. Unmixed CCA sequence VFD. This is based on a sequence realization of 10 million. The last ten binary orders of magnitude in the computation are displayed.

attractor used through this research where it is shown that it, in fact, is a space-filling function by the value of two obtained in the fractal dimension considered. The variance fractal dimension of the chaotically mixed sequence is presented in Fig. 5.4 where it is seen that the same level of fractality is preserved, but the exponent for which the power law holds is higher than the

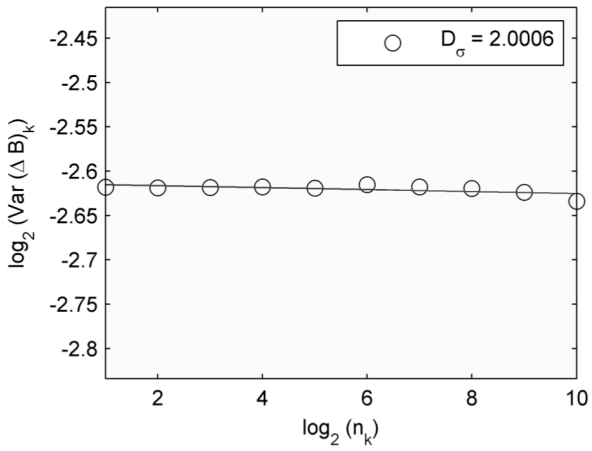


Fig. 5.4. Mixed CCA sequence VFD. This is based on a sequence realization of 10 million. The last ten binary orders of magnitude in the computation are displayed.

unmixed realization of the CCA presented before. Both cases resemble a time series with white noise properties as seen in the previous chapter section two. Continuous-interval cellular automata have the advantage of being capable of exhibiting chaotic behaviour in comparison with GWN or pseudorandom number generators. Also, one advantage over true random number generators is that there is no need to store the realization of the deterministic sequence or transmit it by using a secure channel to further use it to encrypt or decrypt the data. Gaussian white noise sequences are not deterministic. Also, TRNGs capable of generating non-periodic sequences of numbers are not easy to design. If true random number generators are considered, extra hardware and software is added to a cryptosystem based on them. This extra equipment also brings possible flaws that could be used to cause a possible exploit in such cryptosystem.

5.3 Spectral Fractal Dimension

Given that a time series can be stationary but not necessarily present all the frequencies in its power spectrum, this section is dedicated to test the cryptosystem power spectrum exponent. The

method considered for this is presented and verified in chapter four section three.

In order to test the cryptosystem it is necessary to have a stationary time series realization. From the previous section, it is demonstrated that the cryptosystem is FSS10 stationary for a window length of at least $n_w = 2^9$ samples as seen in Fig. 5.1b.

Figure 5.5a shows the PSD for the realization of a time series yielded by the cryptosystem with size $n = 2^9$ while Fig. 5.5b displays its PSD with Spanish moss effect reduction. In figure

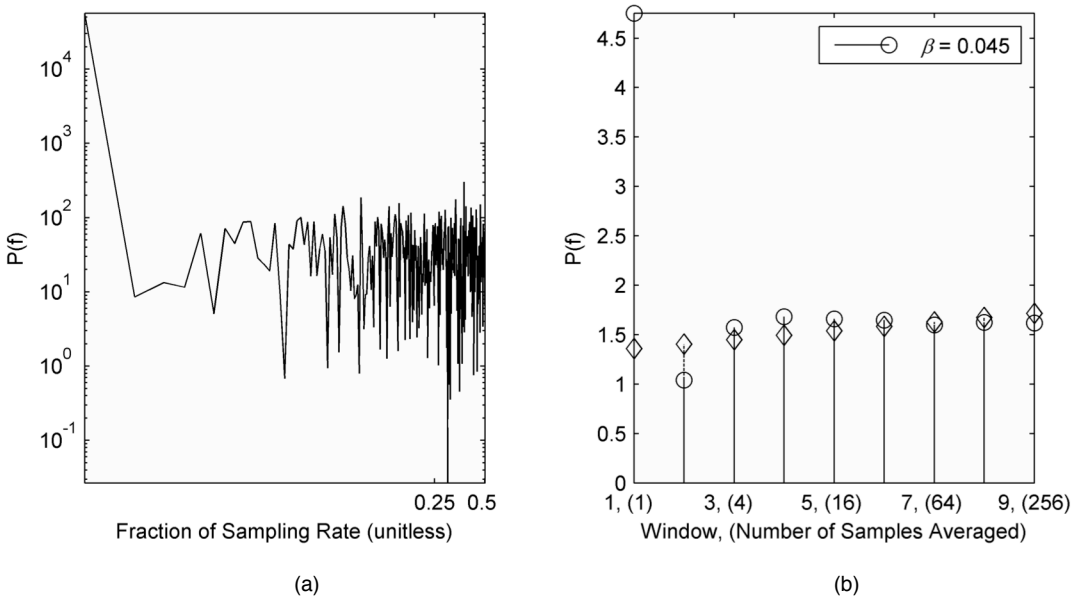


Fig. 5.5. **(a)** Power spectrum density of a time series with size $n = 512$ produced by the cryptosystem. **(b)** Power spectrum density with Spanish moss reduction.

5.5a it is shown that all frequencies are present in the PSD and figure 5.5b shows the line obtained by the weighted least squares fitting calculations. The power spectrum exponent is formally defined as $\beta \sim 0$, which is a really good characteristic within the cryptosystem. It then becomes manifest that the cryptosystem has similar characteristics than those of white noise. The significance of this section is that the smallest stationary window was chosen to implement the

test, which implies that bigger windows fulfill the stationarity testing FSS10.

The cryptosystem complexity has been tested using the VFDT and SFD after extensively verifying their implementations. It has been found that the cryptosystem complexity is the same as a space-filling function similar to the characteristics of white noise. Based on this, the cryptosystem provides a high scrambling level to protect data in secure communications systems.

5.4 Surrogate Data

This subsection explores randomly shuffled surrogates as an endeavour to strengthen the cryptosystem. Chapter four subsection four demonstrates how using this type of surrogates can conceal a chaotic attractor taking as a reference the Hénon attractor. Here, it is particularly intended to reinforce the cryptosystem core and rise its security degree to a whole new level by implementing the same idea in the multi needle attractor.

Classic chaos-based cryptographic systems present vulnerabilities like: (i) identification of a chaotic system in the time domain, (ii) model reconstruction by monitoring state variables, or (iii) synchronization search. The presented cryptosystem is considered secure because it implements chaotic mixing of short chaotic sequences, making the cryptosystem already capable of resisting the vulnerabilities previously mentioned. Now, on top of that, the idea of using a technique to conceal the chaotic attractor reinforces the core of the cryptosystem for robustly supporting against attacks attempting to break it. The chaotic attractor used in the cryptosystem is shown in Fig. 5.6 in its canonical form. After testing the technique based on surrogate data, it is shown in Fig. 5.7 that no traces of the needles are left. The attractor reconstruction is done using (4.22) and $m = 3$. When comparing the severe differences in Figs. 5.6 and 5.7, it can be stated that a needle was hidden in a haystack, but it has now vanished.

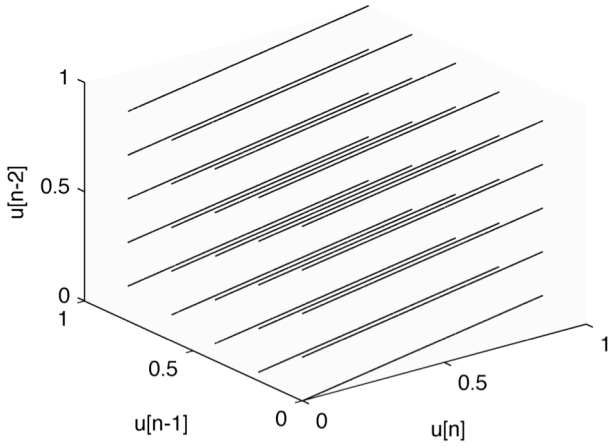


Fig. 5.6. Multi needle strange attractor.

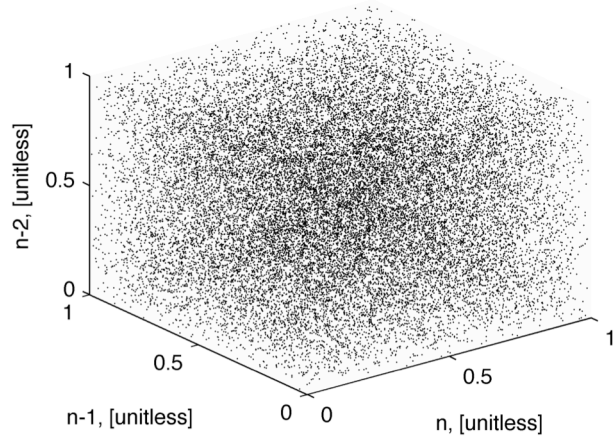


Fig. 5.7. Multi needle attractor concealed by using randomly shuffled surrogates.

Four different methods for testing the cryptosystem are performed in the research presented by this thesis. The outcomes obtained across (i) FSS10, (ii) VFD, (iii) SFD, and (iv) surrogate data provide positive implications for the degree of security offered by the cryptosystem. The implementations of these methods are carefully verified by using known signals, which offers high confidence to the quantitative measures obtained through them.

By using surrogate data, it has been demonstrated that the chaotic attractor of the cryptosystem is concealed. Thus, making it extremely difficult for an attacker attempting to break the cryptosystem. The surrogates reinforce the cryptosystem by providing an extra protective layer. Through this layer, the chaotic like properties of the cryptosystem are completely concealed making it untraceable for attacks based on (i) time domain identification, (ii) system state monitoring, and (iii) synchronization. Surrogates also provide white noise-like properties for the cryptosystem chaotic attractor because one value appears to have no correlation to the others in its cryptographic sequences. Even though the cryptosystem has been found to exhibit properties like those of white noise, as seen across the outcomes from the VFD and the SFD, chaotic signals are known to be deterministic. This is a challenge posed by chaos phenomena and it has been attacked in two flanks: firstly, the use of chaotic mixing and finally, through surrogates. Since the type of surrogates, randomly shuffled, do not change the statistical moments in a time series, the uniform distribution that the cryptosystem has is not altered and thus the cryptosystem does not lose its capabilities of resisting statistical based cryptanalysis.

5.5 Summary

The cryptosystem proposed has been tested in this chapter by using four different signal analysis methods capable of producing quantitative measures: (i) the stationarity map, based on the FSS10 method, precisely provides the minimal window size in which the cryptosystem is considered FSS10 stationary; (ii) by implementing the VFDT and the SFD, it is concluded that the cryptosystem has highly complex behaviour and its characteristics are comparable to the ones of white noise; and through the use of (iii) surrogate data the capability to conceal the cryptosystem chaotic attractor has been revealed, providing the cryptosystem with stealth

capabilities and demonstrating that its chaotic attractor cannot be reconstructed by exposing state variables, attempting to build a model of the system, or finding synchronization patterns. This sustains that the data protected by this cryptosystem are secure and represents a strong wall to attackers. Four encryption experiments that use the high quality cryptographic sequences that the cryptosystem is capable of providing are presented in the subsequent chapter.

CHAPTER VI

DESIGN OF EXPERIMENTS

6.1 Introduction

After extensively validating the implementations of the algorithms in the verification chapter, it has been confirmed that the methods used bring the proper outcomes when processing known signals. The referred methods being: FSS10, VFDT, SFD, and random shuffled surrogates, while the signals used are: white noise for the first three cases and the Hénon attractor in the final case.

The preceding chapter is dedicated to significantly testing the cryptosystem applying the same methods developed in the verification chapter. The quantitative measures obtained from this endeavour sustain that the cryptosystem: is stationary in the strong sense, it resembles similar characteristics to white noise, and its chaotic attractor can be concealed to bring its security to a higher degree.

This chapter describes accurately the experimental environment available. The details about the design of the experiments are also described. This description serves as a preamble to execute the experiments. The experiments execution is addressed in the next chapter where some of the cryptosystem potentials are disclosed.

6.2 Experimental Platform

The hardware employed consists of a laptop (Fujitsu LifeBook A Series) counting with an Intel Centrino Duo microprocessor and 1 GB in RAM. This laptop has Windows 7 as operating system and Matlab using as simulation environment. It is important to note that if dynamical

systems defined by differential equations were used, instead of CCA in this limited computer, it would require weeks of processing time to obtain useful sequences to conduct similar cryptographic experiments. Execution time is one feature where CA demonstrates some of their powerful advantages.

6.3 Cryptosystem Setup

The cryptosystem mathematical basis has been introduced in chapter two subsection two. This foundation is described by the equation (2.2). As described in chapter three where the cryptosystem is unveiled, there are two distinct groups with modules labeled as W and WRA. Each group contains 10 modules. Both groups W and WRA, and the SG include an instance of equation (2.2) in them.

From the equation (2.2), the specific values for the parameter g and IC are setup as follows: $g = 3$ in the case of the SG, $g_{W_1} = 5, g_{W_2} = 7, ..., g_{W_{10}} = 23$ for the W and $g_{WRA_1} = 25, g_{WRA_2} = 27, ..., g_{WRA_{10}} = 43$ for the WRA. The initial condition value for the SG is $IC = 0.0802$ in all cases.

6.4 Experiments Design

Having the knowledge that the cryptosystem is capable of protecting information securely, as justified by the outcomes from the tests through which the cryptosystem has been exposed, the lines included in this section address the set of experiments selection and their design to visualize the cryptosystem performing encryption/decryption processes with data available in the real world.

The approach considered for the experiments selection demonstrates the cryptosystem

capabilities to secure data from different sources. The experiments designed have been carefully selected to show this powerful feature of the cryptosystem. Also, the four experiments have been selected as the minimalistic set to conduct this demonstration. The experiments design consists of the following cases:

6.4.1 Text String

A text string having a periodic pattern is considered for this case. The reason behind this is to explore if periodic patterns are shown in the CT after encrypting the text string. The text string, 19 millions samples long, is composed by the repetition of the arbitrary characters YAGGHGTLVJDTGH.

6.4.2 Sound Wave

The ciphering of a sound wave is proposed as an experiment to explore if the information contained in auditive form could be made unintelligible to the human ear. This experiment contemplates transforming both channels of a song in stereo format. To analyze this properly, the final result of this experiment should be delivered in the same dynamic range as the original waveform.

6.4.3 Grayscale Image

The processing of visual information by the cryptosystem is also proposed. In this case an experiment consisting of encrypting a grayscale picture is devised. This experiment should use an image having periodic patterns. Once the image representing the PT has been transformed into CT, the CT should be analyzed for traces of the original information left in the CT after the encryption stage has been done.

6.4.4 Color Image

The last experiment devises the encryption of a color picture. The particular objective is to explore if similar results to the grayscale image processing are obtained after it is encrypted. Also, this experiment is proposed because it is desired to explore if the information contained in the images could be completely vanished and if no visible patterns are left once the plaintexts are transformed by the cryptosystem.

6.5 Comparison of the Cryptosystem with Alternative Encryption Schemes

In order to have a reference point about how the cryptosystem performs, it is necessary to compare it with known encryption schemes. The cryptosystems selected for comparison are RSA, ElGamal, and the use of a stream of radio background noise. The word length to secure the data has been selected for 2^{16} bits. The decision to select this word length is based on the limitations of the equipment available for comparison of the cryptosystem considering the resources required for the public key cryptosystems implementations. These implementations performed in non-dedicated hardware, but through simulations in Matlab, provide an insight on how the cryptosystem presented in this thesis could compare to others. This structure brings the same resources and limitations for all the cryptographic schemes compared.

The implementation of the RSA algorithm is based on the proposed by Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone [MeOS01]. The cryptosystem is compared to the alternative cryptosystems mentioned previously using the grayscale picture to have a visual reference framework. This framework allows for (i) identification of patterns in the ciphertext, (ii) speed of encryption/decryption, (iii) FSS10 analysis of ciphertext, and (iv) analysis of the ciphertext complexity by using the VFDT and the SFD.

6.6 Summary

The linkage between verification, testing, and experiment design is provided. The research presented in this thesis is coherent in these three stages. The important feature of the cryptosystem, which is capable of protecting data from different sources, is highlighted. The details about the cryptosystem setup are provided. The intentions of the set of experiments are also carefully stated in a simplistic manner. A comparison framework of the cryptosystem presented with alternative cryptographic schemes is devised.

CHAPTER VII

EXPERIMENTAL RESULTS AND DISCUSSION

As discussed in the previous chapter, the cryptosystem based on CCA that has been proposed is implemented in a very limited hardware platform. However, this equipment is enough and capable of illustrating the different encryption and decryption experiments previously designed. This chapter explores the actual outcomes when processing data from different sources with the purpose of protecting them cryptographically.

Four different sources of data are considered in this thesis to show the wide areas of applications that the MMDC can reach and the flexibility for its implementation. In all the cases presented, no information loss took place during the encryption and decryption processes. The main goal of cryptography (*i.e.*, to keep the information intact and hidden) has been demonstrated. A similar degree of protection can be achieved in all the four experiments performed. This provides an intuitive understanding of how versatile the MMDC is for robust cryptographic applications. Thus, any entity that needs to secure or protect information could benefit from the advantages of the MMDC.

A desirable property of any cryptosystem is that a small change of one bit of the key should produce a change in many bits of the CT [Nich99]. This is known as “avalanche effect” and is present on most of the modern cryptosystems in different degrees according to the method in which they mix or “scramble” the bits. The proposed cryptosystem inherently achieves this objective because a small change introduced in the initial conditions of a chaotic system produces a completely different sequence ([Kins11a], [PeJS04], and [Stro00]).

Applications in both symmetric and asymmetric operation schemes are possible with the presented cryptosystem if appropriate protocols are used for key exchange. The symmetric case can follow the classical protocol where the key is distributed through a secure channel. The *Menezes-Qu-Vanstone* (MQV) protocol widely used in ECC can be implemented for the asymmetric case [Rosi99]. This protocol avoids the "*man in the middle*" (MIM) attack by "perfect forward secrecy" and performs authentication of the key holders [Rosi99].

Since the length of the encryption sequences used is the same as the different PTs used, this cryptosystem exhibits "perfect secrecy" according to Shannon ([Assc06], [Nich99], and [Shan49]). This condition ensures that the uncertainty of the CT is maximized and theoretically the cryptosystem is capable of resisting cryptanalysis even in the presence of infinite computing power [Nich99]. Consequently, the presented cryptosystem is a *one-time pad* where interception of any amount of CT is not sufficient for an attacker to break it [Nich99]. This theoretically nullifies any effort to reconstruct the PT via any or the following cryptanalytic methods: horizontal or lengthwise analysis, cohesion, reassembly via Kasiski or Kerckhoff's columns, repeats, or internal framework erection [Nich99]. However, since no cryptosystem can be considered canonically secure (*i.e.*, quantum computers and quantum memories used to extract photons from light pulses for further analysis in QC attacks [Assc06]), this thesis does not attempt to claim that the cryptosystem presented is so. Cryptosystems succeed if the amount of processing power, time, or other resources that could be exploited in attempting to break them requires a very long time (*i.e.*, exceed the human life span). The attempt to break a cryptosystem under such conditions makes it unsuccessful. Successful attacks against *one-time* pad cryptosystems must be against the method used to generate the encryption (*i.e.*, the uniformly

pdf from a pseudo random number generator) sequence itself [Nich99]. This, possibly, is one of the most difficult challenges posed for an attacker. Should an attacker succeed in characterizing the system, then brute force attacks might bring out the true PT, but it also yields every other possible PT of the same length. Random transposition or substitution seems to offer a very high level of security, because it should be impractical for an enemy interceptor to unscramble even a short message [Sing99]. It should be noted that the number of possible solutions increases as the PT lengthens, and rapidly reaches the point where more computing power and a higher amount of time are required to come up with all possible solutions of an attack of this nature [Nich99]. Long encryption sequences to encrypt a PT can be obtained also by means of QC requiring highly specialized equipment [Assc06].

Attacks to cryptosystems based on chaos can succeed under the following conditions: (i) identification of the chaotic system in the time domain by one of its state variables, (ii) recovering the message signal by using an approximate model with some errors which can be easily removed by standard filtering methods (*i.e.*, by means of sample vectors dictionaries [RoNB08]), or (iii) finding the synchronization used between the encrypter and decrypter [YaWC97]. Cryptosystems or communications schemes that use synchronization (*e.g.*, [CuOp93] and [RaRo07]) usually transmit a variable between the encrypter and decrypter under the assumption that it is not noticed. These implementations involve normally at least one analogue component used to develop chaos phenomena, and are used mainly for audio transmissions “masking” a PT. In cryptography based on synchronized chaos, the encrypted signal is obtain by adding the lower amplitude PT to the higher amplitude chaotic signal [Assc06]. In this method small errors during decryption can be either tolerated, or are not

noticeable. Due to a short time of a live communication, schemes like these can be considered secure up to some degree, but considering the weaknesses presented above, neither long term data security, nor data integrity are provided. Synchronization between the encrypter and decrypter in the proposed cryptosystem is strictly forbidden and just the final product of encryption is considered secure for storage or transmission. This avoids exposing state variables and, as a consequence, the door to reconstruct a model of the system based on monitoring one of its state variables is closed.

7.1 Experiment 1: Text

A plaintext formed by the repetition of the arbitrary characters YAGGHGTLVJDTGH to form a periodic sequence approximately 19 millions in length is shown in Fig. 7.1.

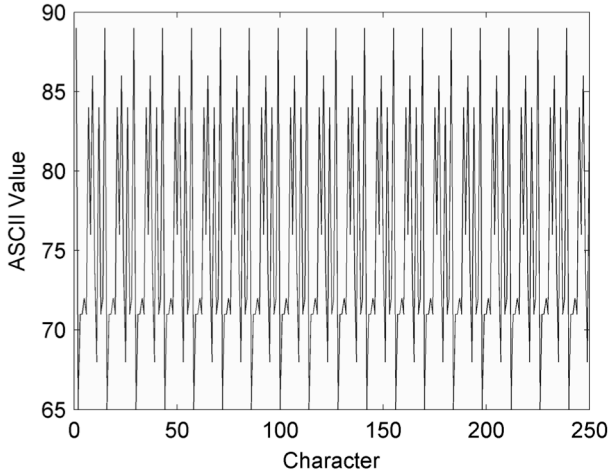


Fig. 7.1. Plaintext sequence. Just the first 250 Characters are shown. Approximately 19 million characters where generated.

An encryption sequence or key of the same length is provided by the SG applying (2.2) and then

using Eq. (3.2) the PT is ciphered. The first 250 ASCII codes of the CT are shown in Fig. 7.2

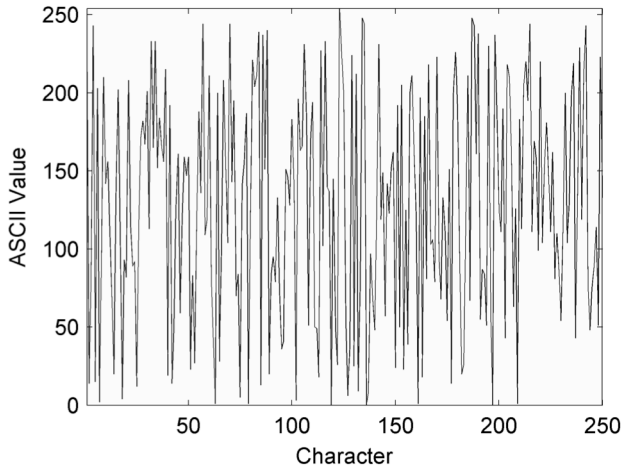


Fig. 7.2. Sequence of encrypted ASCII code. Just the first 250 ASCII codes are shown. Approximately 19 million characters where ciphered.

where no obvious repetitive patterns exist even though a periodic pattern exists in the PT. The characters in the CT vary through all the possible ASCII code values.

7.2 Experiment 2: Sound

An arbitrary song (with approximately 13.5 million samples in the right and left channel) is chosen as PT. A key of 27 million samples in length is generated by the SG to encrypt the information in both channels of the stereo audio. The original information amplitude varies between -1 and 1. The information is ciphered by changing the values to a range from 0 to 2 as minimum and maximum respectively, then (3.2) is applied, and finally the original values are restored to -1 and 1.

Figures 7.3 to 7.6 show 2,000 samples where the plain texts are sinusoidal waveforms found in stereo speech. The ciphertexts are the encrypted left and right channels resembling a random

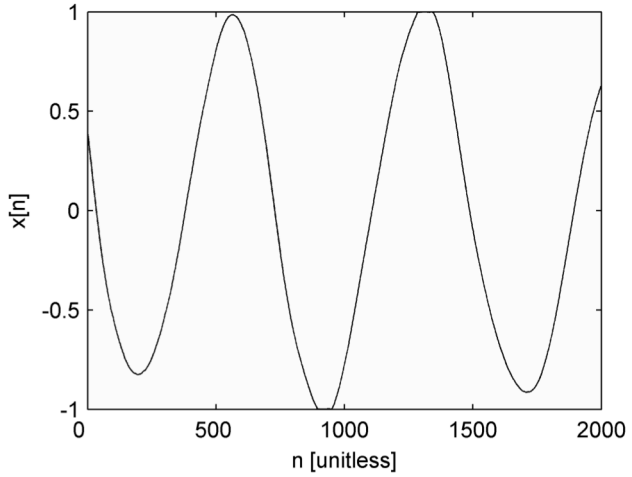


Fig. 7.3. Left channel segment from a stereo sound signal containing approximately 13.5 million samples. Just two thousand samples are shown.

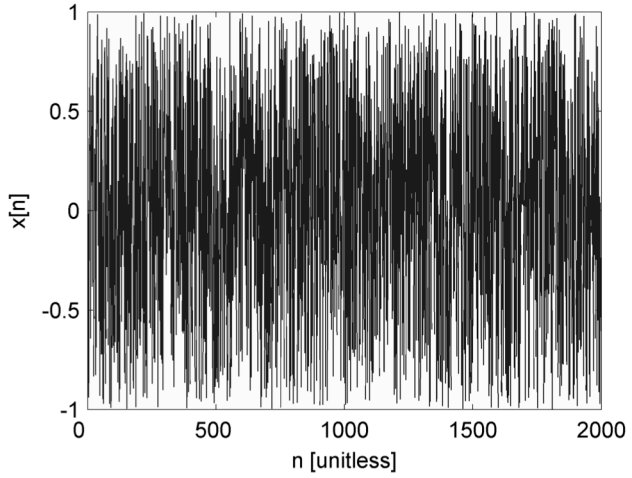


Fig. 7.4. Encrypted left channel segment from a stereo sound signal containing approximately 13.5 million samples. Just two thousand samples are shown.

sequence. A given sample takes any value in the corresponding dynamic range producing a sequence with incomprehensible meaning when it is reproduced. In fact, when one listens to this encrypted sequence it resembles a strong rainy storm regardless of the original information

contained in the original sound sequence.

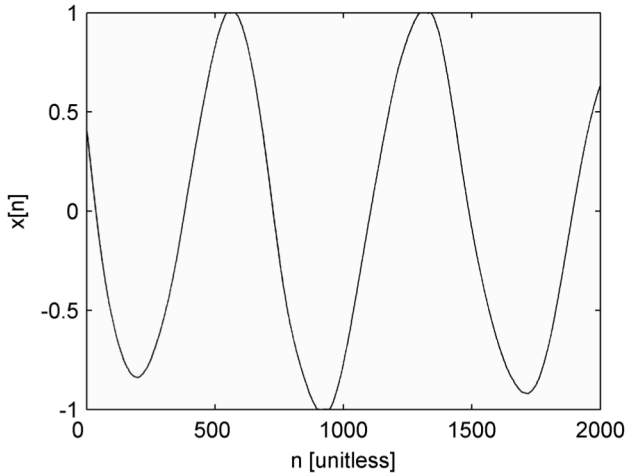


Fig. 7.5. Right channel segment from a stereo sound signal containing approximately 13.5 million samples. Just two thousand samples are shown.

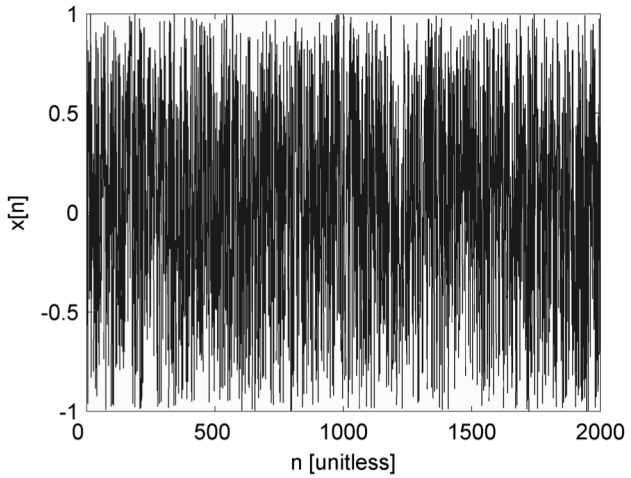


Fig. 7.6. Encrypted right channel segment from a stereo sound signal containing approximately 13.5 million samples. Just two thousand samples are shown.

7.3 Experiment 3: Grayscale Image

Pictures in grayscale and colour are also included in the scope of the cryptographic experiments designed in the previous chapter. For the grayscale case, the picture of Fig. 7.7 is

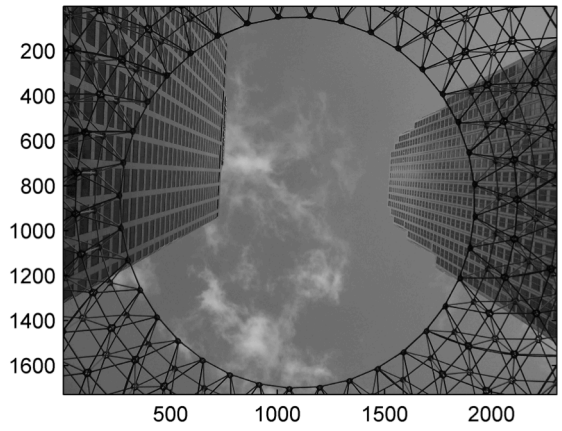


Fig. 7.7. Grayscale picture 1,728×2,304 pixels (width×height) of some buildings at downtown in Los Angeles, CA, USA.

acting as PT. An encryption key approximately 4 million in length is generated to cipher the information contained in the picture of 1,728×2,304 pixels (width×height). The encryption sequence is reshaped to get a matrix the same size of the original picture and then by applying (3.2) the ciphered picture presented in Fig. 7.8 is obtained.

7.4 Experiment 4: Color Image

Figure 7.9 presents a similar process done to the colour picture with the slight difference that it is a three-dimensional array 1,728×2,304×3 (width×height×colour intensity) that contains the values for the red, green and blue (RGB) components. A sequence approximately 12 million in length is generated and reshaped into a three-dimensional array the same size mentioned before, and by the use of (3.2) the colour picture is ciphered as illustrated in Fig. 7.10.

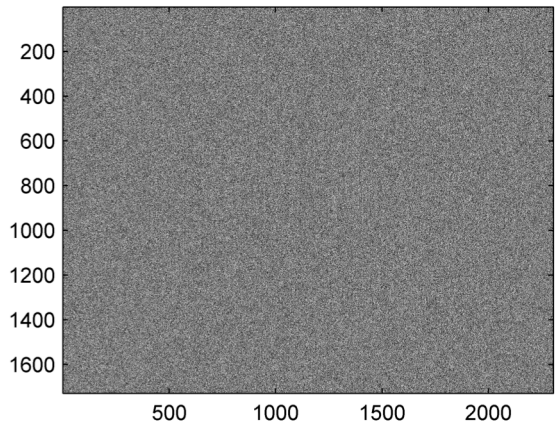


Fig. 7.8. Grayscale encrypted picture 1,728×2,304 pixels (width×height) of some buildings at downtown in Los Angeles, CA, USA.

Fig. 7.9. Colour picture 1,728×2,304×3 pixels (width×height×colour intensity) of Houdini's star in the Hollywood Walk of Fame at Los Angeles, CA, USA.

Both cryptographic experiments involving images show no traces of the original information in the original grayscale and colour pictures in the corresponding CTs after the encryption process has taken place.

The average encryption/decryption processing time was 23, 81, 6, and 13 seconds for the text, music, grayscale picture, and colour picture respectively. The text, grayscale picture, and

colour picture used vectors capable of storing 8 bits data. The audio case used a vector with data storage capacity of 64 bits. The processing times obtained with the MMDC are highly competitive against the performance of alternative cryptosystems (*i.e.*, RSA16, ElGamal, and a cryptosystem based on radio background noise), as presented in the following section.

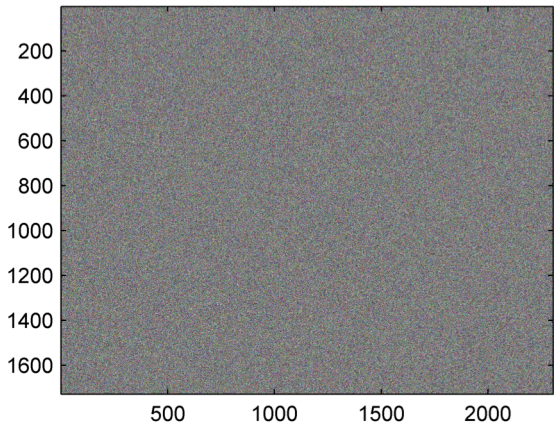


Fig. 7.10. Encrypted colour picture 1,728×2,304×3 (width×height×colour intensity) pixels of Houdini's star in the Hollywood Walk of Fame at Los Angeles, CA, USA. A sequence approximately 12 million in length was used.

7.5 Analysis of the Experiments Histograms

The histograms of the PTs and CTs are presented in Figs. 7.11 to 7.16 and 7.17 to 7.22, respectively. The histograms of the PTs for the text string, the sound wave, the grayscale picture, and the different RGB components in the colour picture show diverse shapes. It is seen that the information after encryption has a uniform distribution even though different sources as PTs are contemplated. The figures showing the CTs histograms are configured to display signal amplification. This allows looking for specific signal characteristics as hooks. All the CTs histograms have a variability range that changes in a scale of thousandths as shown in the Figs.

7.17 to 7.22. A cryptosystem that exhibit good uniform distribution characteristics after ciphering the elements of the PT provides high resistance to statistical analysis [YiRY09].

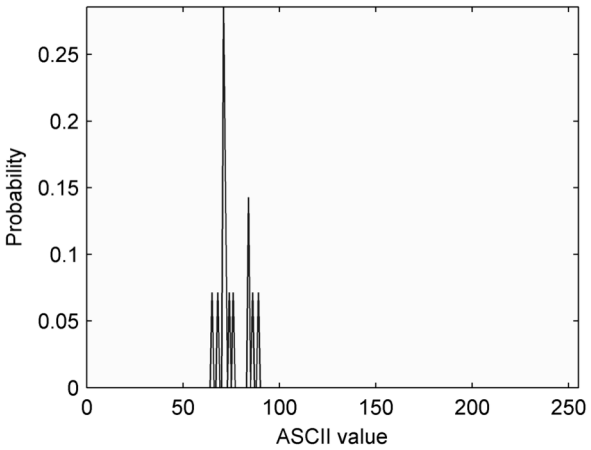


Fig. 7.11. Original text histogram. The number of characters considered are approximately 19 million.

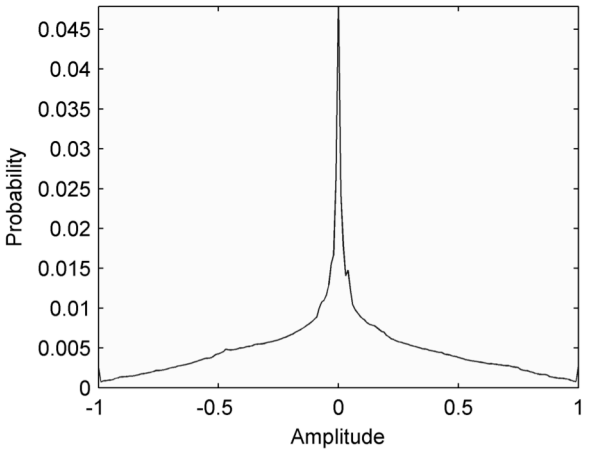


Fig. 7.12. Original sound signal histogram. The number of samples considered are approximately 27 million.

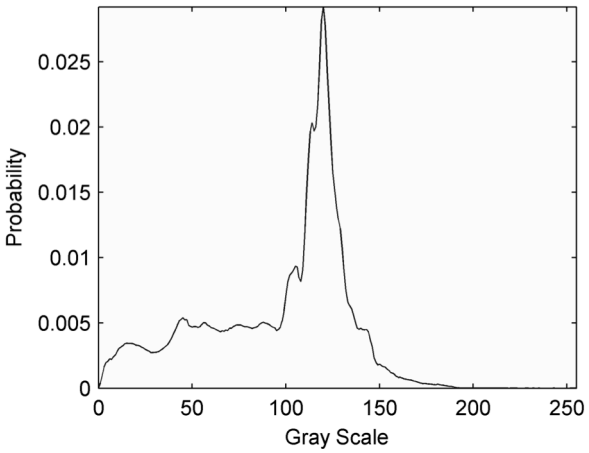


Fig. 7.13. Original gray scale image histogram. The number of pixels considered are approximately 4 million.

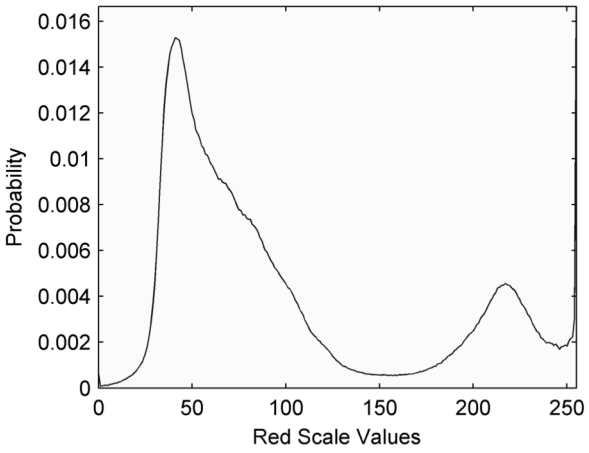


Fig. 7.14. Original color image histogram illustrating the red component. The number of pixels are approximately 12 million.

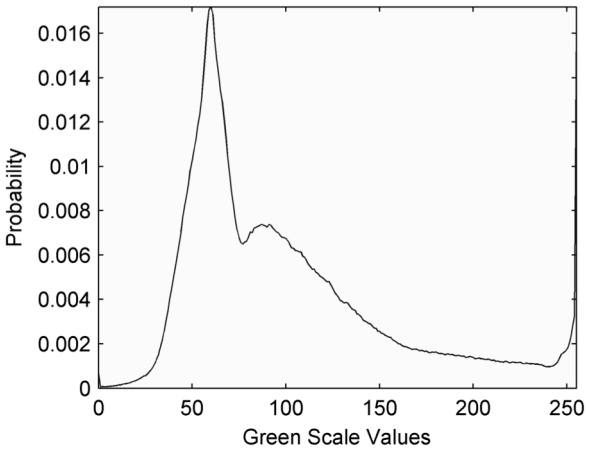


Fig. 7.15. Original color image histogram illustrating the green component. The number of pixels are approximately 12 million.

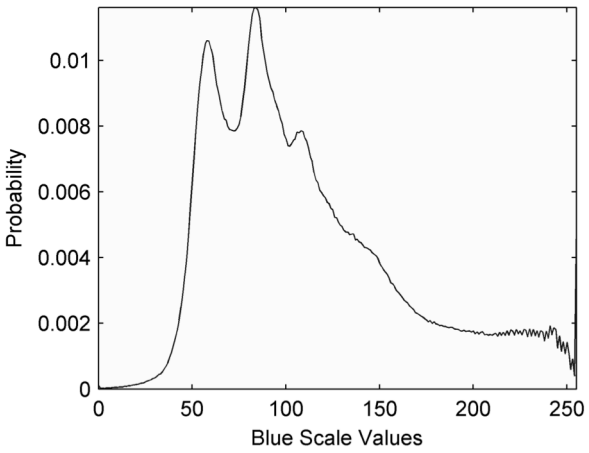


Fig. 7.16. Original color image histogram illustrating the blue component. The number of pixels are approximately 12 million.

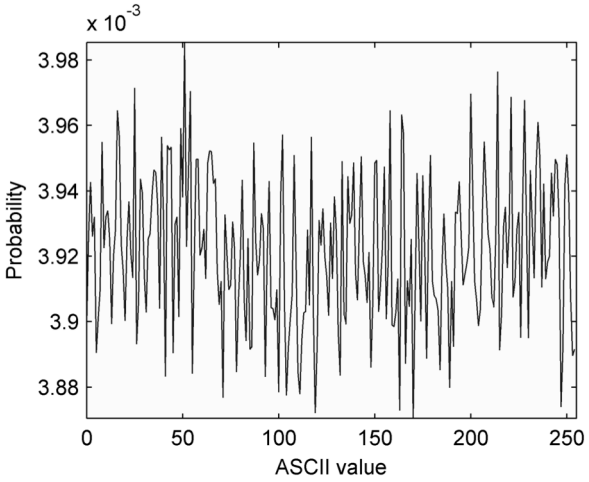


Fig. 7.17. Histogram of the text after encryption. The number of characters are approximately 19 million.

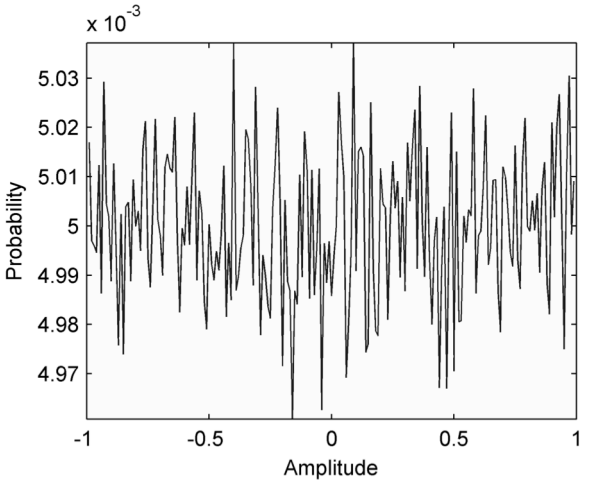


Fig. 7.18. Histograms of the sound signal after encryption. The number of samples considered are approximately 27 million.

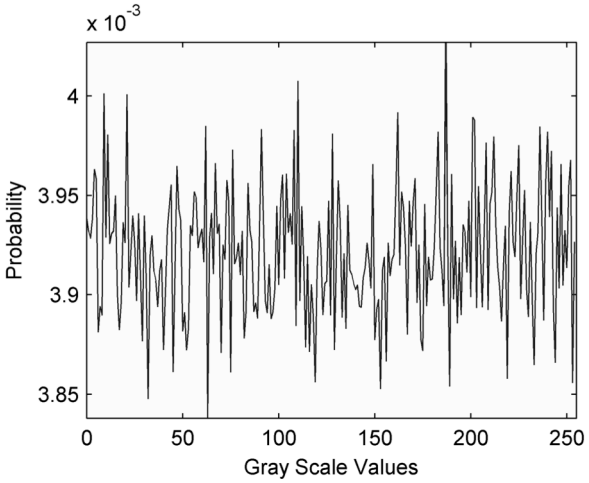


Fig. 7.19. Histograms of the gray scale image after encryption. The number of pixels considered are approximately 4 million.

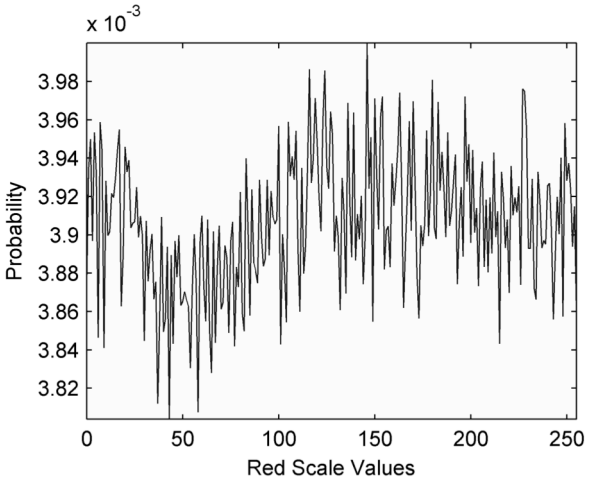


Fig. 7.20. Histogram of the red component in the color image after encryption. The number of pixels are approximately 12 million.

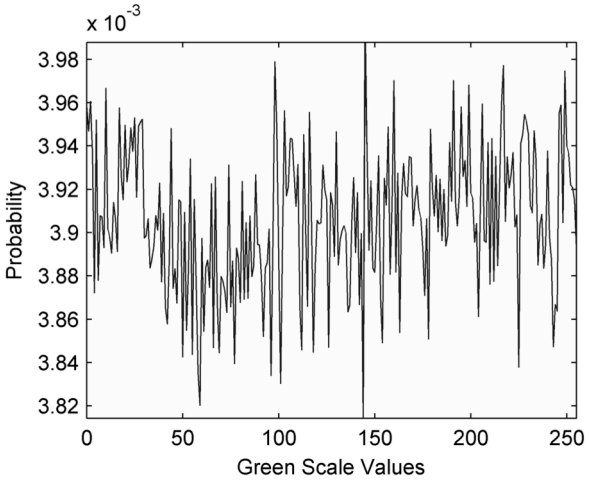


Fig. 7.21. Histogram of the green component in the color image after encryption. The number of pixels are approximately 12 million.

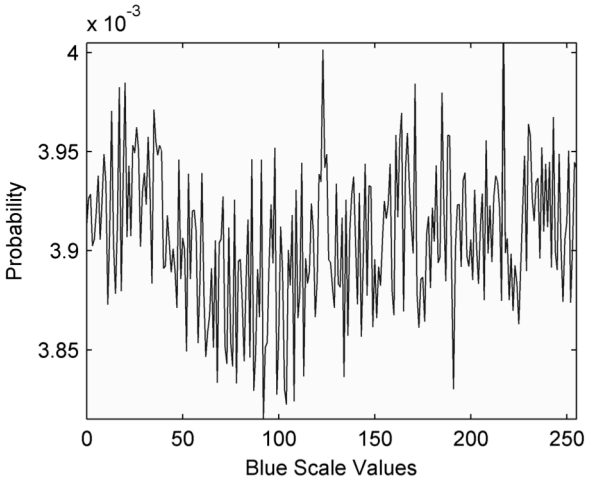


Fig. 7.22. Histogram of the blue component in the color image after encryption. The number of pixels are approximately 12 million.

7.6 Comparison of the Cryptosystem with Alternative Encryption Schemes

The simplest cryptosystem evaluation scheme is to compare the performance of the new cryptosystem with the performance of known cryptosystems. The cryptosystems selected for comparison are (i) RSA ([Gait78], [NBS77], and [RiSA78]), (ii) ElGamal [Kobl94], and (iii) the use of a stream of *radio background noise* (RBN) [Haah12]. The implementation of the RSA algorithm as the proposed by Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone [MeOS01] is used.

The comparison of the algorithm described in Fig. 3.2 with the three encryption schemes is done on a new platform: Matlab 2012a runs in Windows Server 2008 R2 Enterprise 64 bits, 48 GBytes of memory, and the processor is an Intel Xeon Dual Core with a clocking frequency of 2.93 GHz. The different cryptosystem implementations are described next.

7.6.1 Public Key Cryptography Using the RSA Algorithm

The public key encryption scheme RSA, named after its inventors R. Rivest, A. Shamir, and L. Adelman, is the most widely used public-key cryptosystem. It may be used to provide both secrecy and digital signatures, and its security is based on the intractability of the integer factorization problem [MeOS01].

The key generation for RSA PKC revolves around the capability to find two large random (and distinct) primes υ and ρ, each roughly the same size. The modulus η is generated by the product of ρ and υ and $\phi = (\rho - 1)(\upsilon - 1)$. The selection of a random integer ε (*encryption exponent*), $1 < \varepsilon < \phi$, such that $\gcd(\varepsilon, \phi) = 1$. Using the extended Euclidean algorithm to compute the unique integer δ (*decryption exponent*), $1 < \delta < \phi$, such that $\varepsilon\delta \equiv 1 (\mathrm{mod}\, \phi)$. Then, the *public*

key is (η,ε) and the *private key* is φ [MeOS01]. The encryption and decryption rules are defined by

$$e_{K_{RSA16}}[n] = (d_{K_{RSA16}}[n])^{\varepsilon} \bmod \eta \tag{7.1}$$

$$d_{K_{RSA16}}[n] = (e_{K_{RSA16}}[n])^{\delta} \bmod \eta \tag{7.2}$$

Since the length of the test image exceeds the normal plaintexts enciphered by the RSA algorithm, the RSA16 (protection of 2^{16} bits) has been selected. The parameters used for the RSA16 are: $\upsilon = 103$, $\rho = 571$, $\eta = 58{,}813$, $\varepsilon = 22{,}471$, and $\delta = 27{,}211$. The ciphertext of the grayscale picture Fig. 7.7 encrypted by the RSA process is shown in Fig. 7.19. This figure includes the *marginal probability mass function* (mpmf) along the x and y axes.

To have a clearer insight of the shape distribution of the mpmfs, it is necessary to perform a moving average on them. The smoothing algorithm of the mpmfs uses the moving average filter [Smit97] defined by

$$v[n] = \frac{1}{M} \sum_{m=-(M-1)/2}^{(M-1)/2} u[n+m] \tag{7.3}$$

where $u[n]$ is the input signal (*i.e.*, one of the mpmfs along the image x or y axis). $v[n]$ corresponds to the output signal, and M is the number of points (usually an off value) in the average filter. Determining the value for M is not trivial, however, [OHav12] proposes two estimates for it (i) using the ratio between the original signal standard deviation σ_{os} and the target standard deviation after the smoothing σ_{as} (if the signal has features similar to white

noise) and (ii) defining a smoothing ratio approximately equal to 0.01 between the number of samples in the averaging filter M and the number of samples in the input signal $u[n]$. The latter is used to plot the smoothed versions of the mpmfs throughout this paper. This enhanced feature corresponds to the white lines superimposed over the mpmfs in Figs. 7.19 to 7.22. The mpmf along the x axis in Fig. 7.19 is not uniform. Furthermore, the dispersion of the mpmf values is

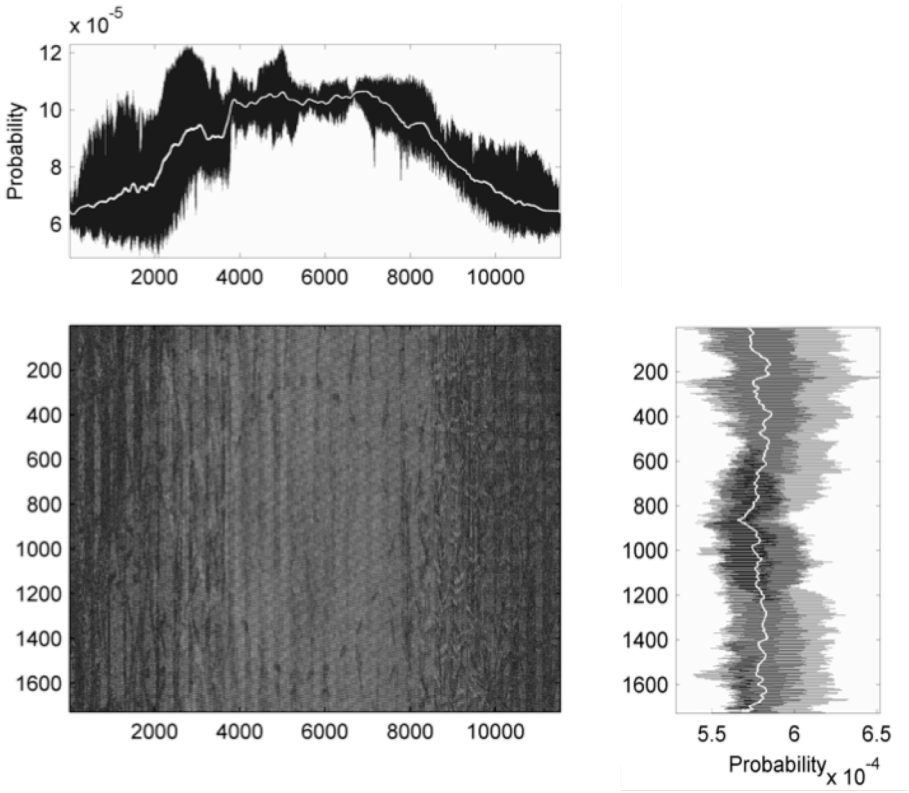


Fig. 7.19. Ciphertext obtained through RSA16. The number of pixels increased from 4 to 20 millions.

not constant with respect to the average values denoted by the white line. The mpmf along the y axis is more uniform in the distribution, and the dispersion is more constant.

The ciphertext also shows that some of the plaintext patterns in Fig. 7.7 reappear very vaguely. The ciphertext size in pixels increases from 4 to 20 millions. This increment is inherent in the RSA algorithm, and could have an impact on computer systems storing vast amounts of data. This is one of the reasons why practical RSA encryption is used most commonly for the distribution of symmetric-key encryption algorithms keys, and for the encryption of small data items [MeOS01]. The encryption time for this experiment is 3,614.69 seconds, while the decryption time is of 480.91 seconds.

7.6.2 Public Key Cryptography Using the ElGamal Algorithm

The ElGamal public-key encryption scheme can be viewed as Diffie-Hellman key agreement in key transfer mode. Its security is based on the intractability of the discrete logarithm problem, related to ECC, and the Diffie-Hellman problem [MeOS01]. The generalized ElGamal encryption scheme is considered for the ECC implementation here [MeOS01].

The key generation for the ElGamal requires creating a public key $(\varphi,\alpha,\alpha^{\varsigma})$ and a private key (ς). This key generation is achieved by finding a large random prime φ and an α of the multiplicative group $\mathbb{Z}_{\varphi}^{*}$ of the integers modulo φ. Finding φ requires specifying its bitlength κ and a security parameter ψ. Finding φ stops when factoring $\varphi-1$ yields a prime factor greater than ψ. Finding α requires choosing a random element in a cyclic group G of order ϖ. A random element (α) from the prime factorization $\varpi=\varphi^{e_1}\varphi^{e_2}...\varphi^{e_k}$ is chosen each time until $\gamma\neq 1$ when $\gamma\leftarrow\alpha^{\varpi/\varphi_i}$ is computed. Once α is defined through this iterative process it becomes an element of the public key set. Selecting a random integer ς, $1\leq\varsigma\leq\varphi-2$, and computing

$\alpha^{\varsigma} \bmod \varphi$ allows to complete the creation of both public and private keys [MeOS01].

In order to make the ElGamal implementation compatible with the RSA, the following adjustments must be made: (i) the security parameter value for ElGamal is chosen as $\psi = 571$ (*i.e.*, the biggest prime factor used in the former), and (ii) the *bit*-length κ is set to 16. Thus, the values of the keys are: public key $(\varphi = 59{,}093,\ \alpha = 3,\ \alpha^{\varsigma} = 47{,}667)$ and private key $(\varsigma = 54{,}015)$. Figure 7.20 shows the encrypted image.

Figure 7.20 shows that the mpmfs are non-uniform along both the x and the y axes.

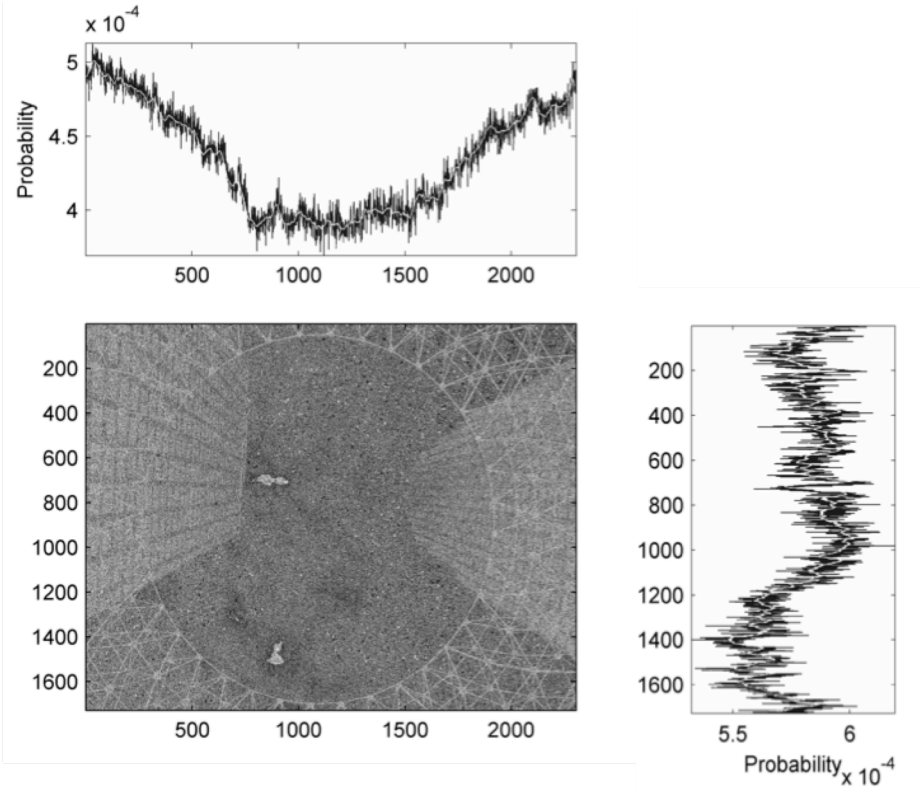


Fig. 7.20. Ciphertext obtained through ECC ElGamal.

Dispersion of the mpmf values is fairly constant and narrow compared to the RSA encoding. Surprisingly, the ciphertext in Fig. 7.20 shows the metallic structure and the buildings more

clearly than in Fig. 7.19. A possible reason that the ciphertext shows those features in the plaintext is that the elliptic curve has not been selected optimally. Encryption/decryption times are much faster (81.94 and 2.43 seconds, respectively) than those for RSA.

The cryptosystems comparison based on mpmfs in Figs. 7.19 to 7.22 provides an insight of the uniformity of the different ciphertexts. The similarities between the mpmf of the cryptosystem using a RBN shown in Fig. 7.21 and the mpmf belonging to the chaos-based cryptosystem depicted in Fig. 7.22 demonstrate that the cryptosystem tested throughout this research has very strong properties for data protection from the statistical point of view.

7.6.3 Stream Based on Radio Background Noise

A source of randomness based on radio background noise [Haah12] is used to encrypt Fig. 7.7. The source of randomness used is also mixed with electronic noise, plasma noise, spatial noise, and deep space noise [Kins07a]. This noise mixture is picked up by a radio receiver interfaced to a sound card of a computer. The mixed noise is digitized using a sampling frequency of 8 kHz and a dynamic range of 8 bits. Several such receivers creating random bit streams are distributed in different geographic locations across Ireland. This distributed configuration merges the random bit streams into a cloud hosting service. This configuration provides increased reliability and performance for the clients (*i.e.*, lottery drawings and general users). The users have access to these bit streams over the Internet. Other services providing a source of randomness (*i.e.*, radioactive decays) also exist on the Internet. The free service used in this research allows a user to receive a data stream of 10,000 elements. A data stream generated on September 11, 2012, is used. The surrogate data technique based on random shuffling that is explained in chapter IV section four is applied to the data stream to get an encryption stream

long enough to match the data contained in the plaintext of Fig. 7.7. A longer sequence with the characteristics of the original RBN sequence is obtained from 400 surrogates.

The equation (3.3) is then modified to

$$e_{K_{RBN}}[n] = \left(p[n] + rbn[n]\right) \bmod r \tag{7.4}$$

where $rbn[n]$ represents the use of a RBN. By applying (7.4) and the source of randomness to

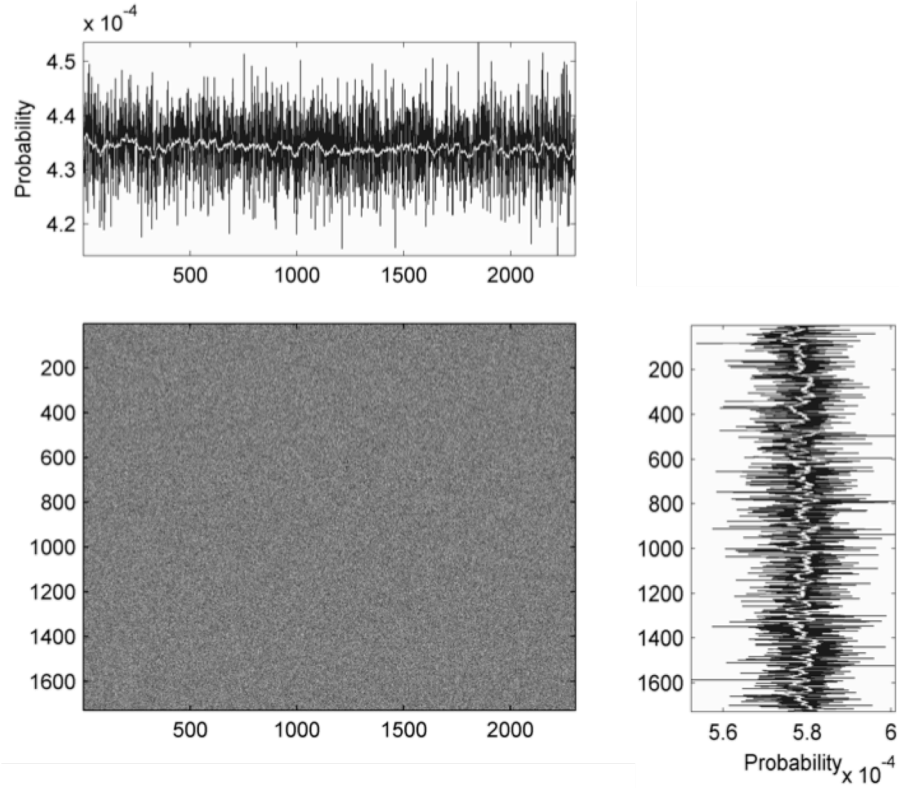


Fig. 7.21. Ciphertext obtained using an encryption stream using RBN.

Fig. 7.7 a ciphertext is generated as shown in Fig. 7.21.

Figure 7.21 shows that the marginal probability mass functions are uniform along both the x

and the y axes. The dispersion of the mpmf values is also constant. Figure 7.21 reveals neither repetitive patterns (as is the case of RSA PKC in Fig. 7.19), nor recognizable features (as in the case of ElGamal PKC in Fig. 7.20). The encryption/decryption times are both similar with a value of 0.14 seconds. This reduction in time is because neither the time required for acquiring the data stream based on RBN nor the time necessary for matching the plaintext length by means of surrogates is considered. The ciphertext in Fig. 7.21 is obtained using RBN and resembles Fig. 7.22, as obtained by the chaos-based ciphertext with a processing time of 0.18 seconds. Notice that Fig. 7.22 is the same as Fig. 7.8, but with the correspondingly mpmfs.

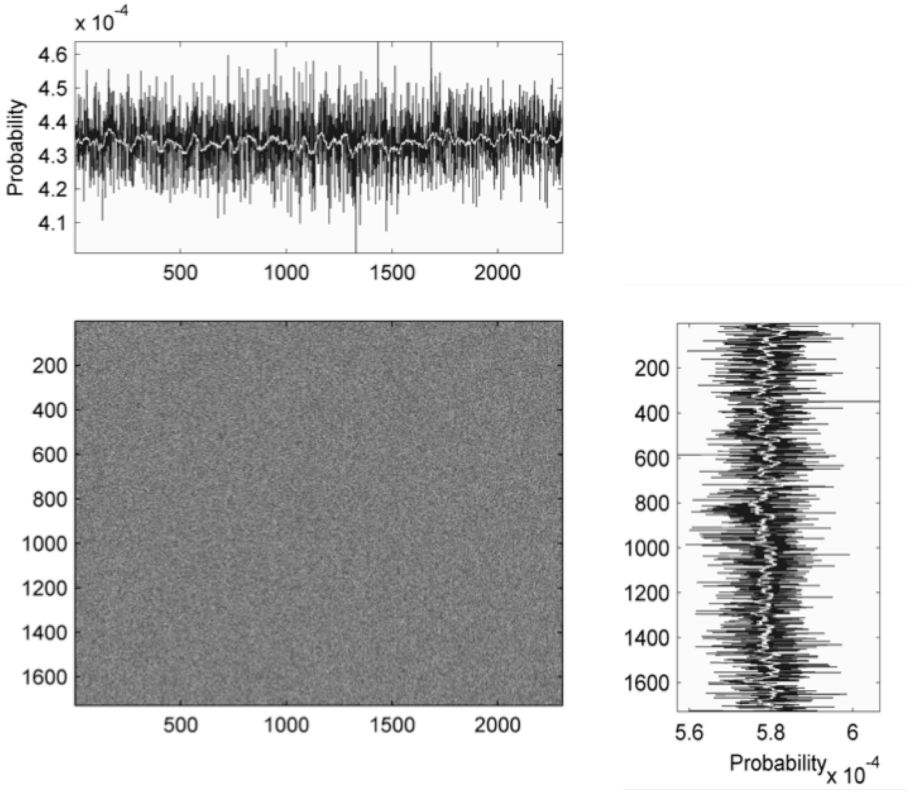


Fig. 7.22. Ciphertext obtained using an encryption stream generated through the chaos-based cryptosystem.

7.6.4 Ciphertexts Stationarity Analysis

The stationary analysis FSS10 presented in chapter IV section one can provide an insight about how the ciphertexts statistical moments vary under different window sizes. Each 2D ciphertext displayed in Figs. 7.19 to 7.22 is converted to a vector of length $n = 245{,}760$, and the window lengths of $n_w = 2^{13}$ to $n_w = 2^5$. Figures 7.23 to 7.26 show the ciphertexts stationarity maps.

Figure 7.23 shows that the RSA16 cryptosystem is non-stationary according to its second statistical moment. The ElGamal cryptosystem is strongly stationary for a window size as small as $n_w = 2^7$ (Fig. 7.24), where the second moment starts to increase prominently. Figure 7.25

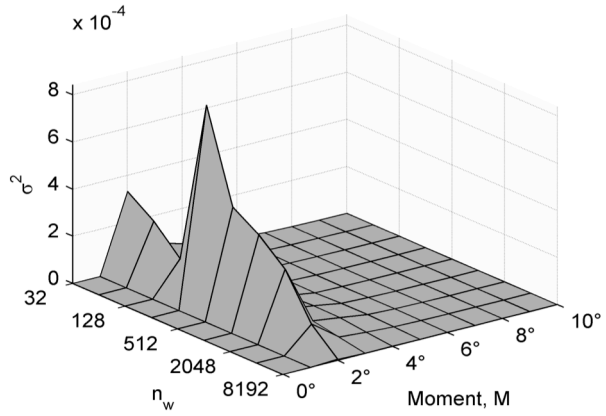


Fig. 7.23. Stationarity map based on σ^2 (variance) of the RSA16 cryptosystem.

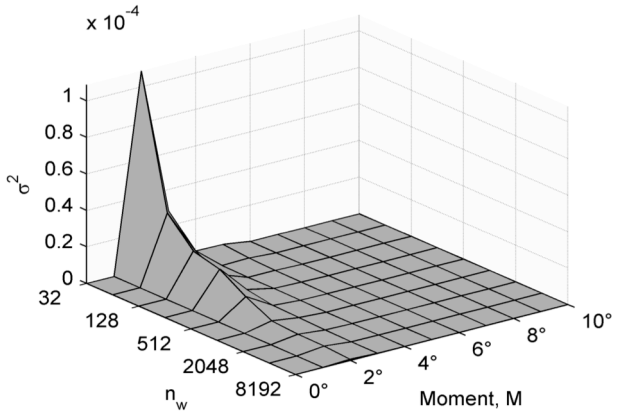


Fig. 7.24. Stationarity map based on σ^2 (variance) of the ElGamal cryptosystem.

depicts the cryptosystem using a RBN, which is strongly stationary for a window as small as $n_w = 2^8$. The chaos-based cryptosystem has a stationarity map that shows that it is strongly stationary for a window size of $n_w = 2^7$ (Fig. 7.26).

The above stationarity tests demonstrate that the ElGamal, RBN-based, and the chaos-based ciphertexts are stationary, but with varying stationarity levels. It is seen clearly in Fig. 7.26 that the cryptosystem based on chaos (i) exceeds the stationarity characteristics of both the RSA16 and the ElGamal cryptosystems (Figs. 7.23 and 7.24), and (ii) its stationarity range exceeds the one of the cryptosystem using a RBN (Fig. 7.25).

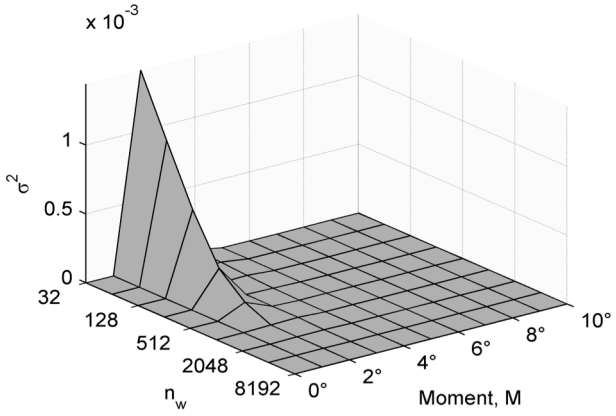


Fig. 7.25. Stationarity map based on σ^2 (variance) of the cryptosystem using RBN.

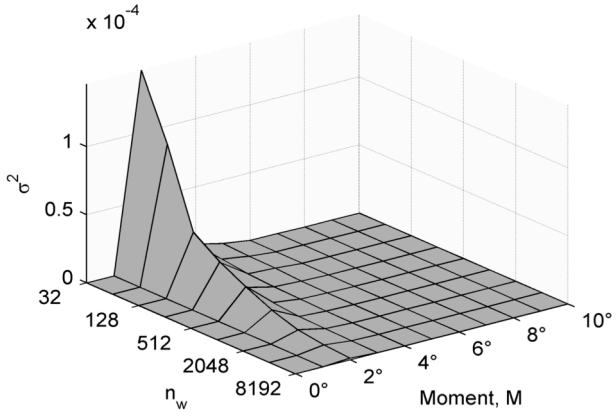


Fig. 7.26. Stationarity map based on σ^2 (variance) of the cryptosystem based on chaos.

7.6.5 Ciphertexts Spectral Fractal Dimension

The final test relates to the complexity of the ciphertext, and is done through a polyscale measure, the (SFD) [Kins11a]. If the value of the SFD, D_β, is close to 2, the ciphertext has the complexity of white noise (a space-filling function). If the value of D_β drops towards 1, the complexity drops that a more correlated ciphertext is obtained. First, a PSD $P(f,T)$ is computed for a ciphertext, and recorded in a log-log scale. Once a section with constant slope is found, it is isolated, and its particular SFD value is then calculated (*i.e.*, $D_{\beta 1}, D_{\beta 2}, ..., D_{\beta n}$). A general SFD $D_{\beta g}$ is also provided for each ciphertext by fitting the

whole set of points in the PSD plot. Figures 7.27 to 7.30 show the SFD analysis.

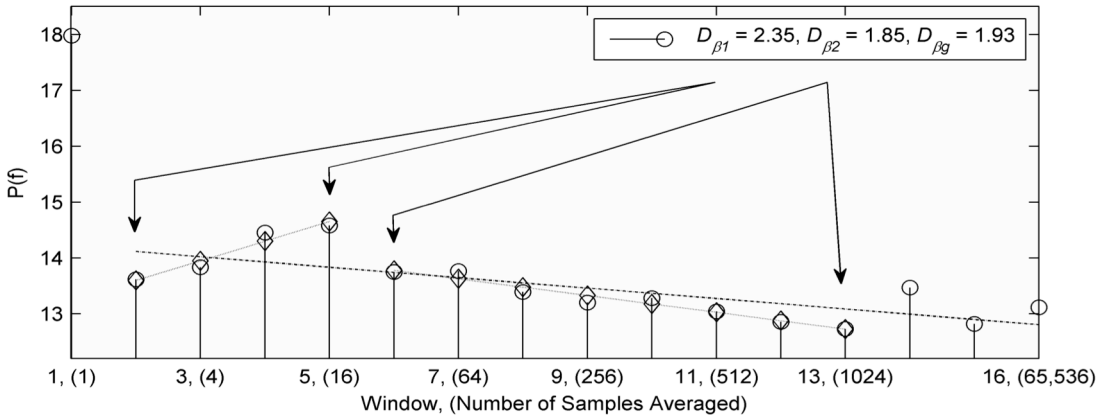


Fig. 7.27. Spectral fractal dimension (D_β) of the RSA16 cryptosystem. Two different fractal sections, $D_{\beta 1} = 2.35$ and $D_{\beta 2} = 1.85$, are seen in the plot. The general SFD is $D_{\beta g} = 1.93$.

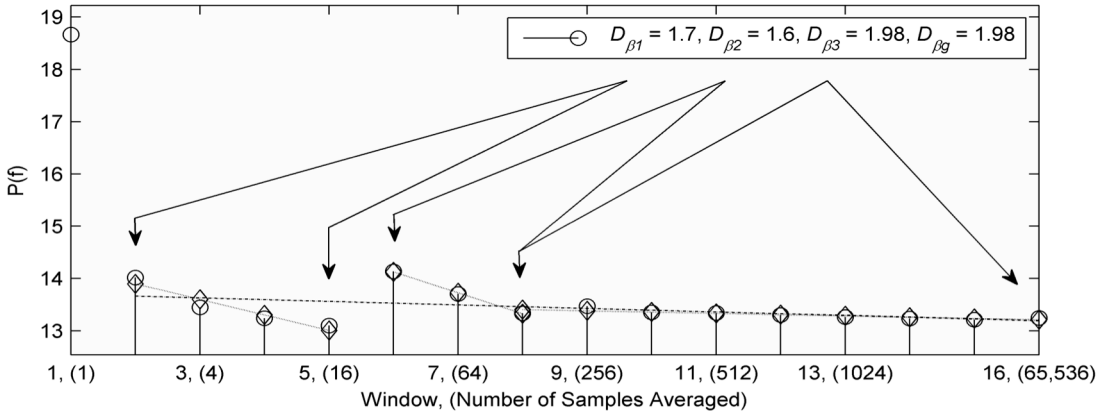


Fig. 7.28. Spectral fractal dimension (D_β) of the ElGamal cryptosystem. Three different fractal sections, $D_{\beta 1} = 1.7$, $D_{\beta 2} = 1.6$, and $D_{\beta 3} = 1.98$ are shown in the plot. The general SFD is $D_{\beta g} = 1.98$.

All figures presented indicate that the ciphertexts contain sections with different SFD values. The RSA16 cryptosystem depicted in Fig. 7.27 has SFD sections that are not close to the desired value of 2. Both the ElGamal (Fig. 7.28) and the RBN (Fig. 7.29) cryptosystems have SFD sections, close to the value of 2 as small as for windows of $n_w = 2^7$ samples. The cryptosystem

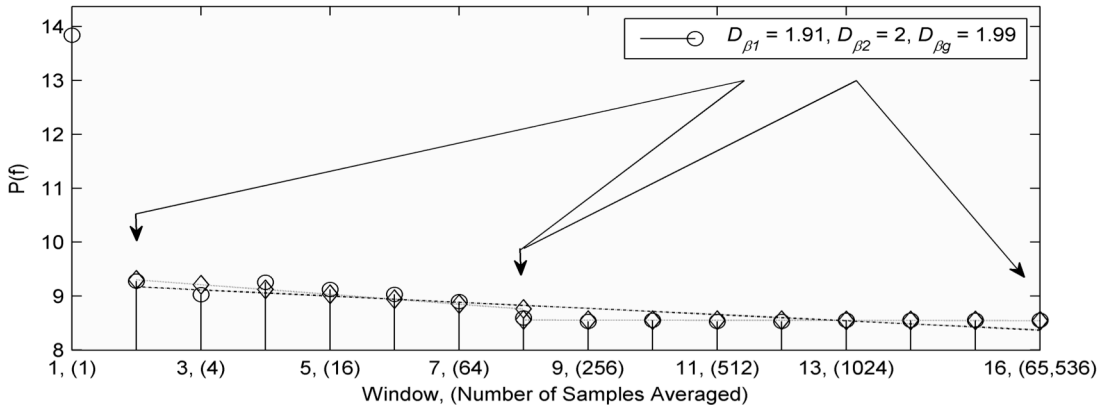


Fig. 7.29. Spectral fractal dimension (D_β) of the cryptosystem using RBN. Two different fractal sections, $D_{\beta 1} = 1.91$ and $D_{\beta 2} = 2$, are seen in the plot. The general SFD is $D_{\beta g} = 1.99$.

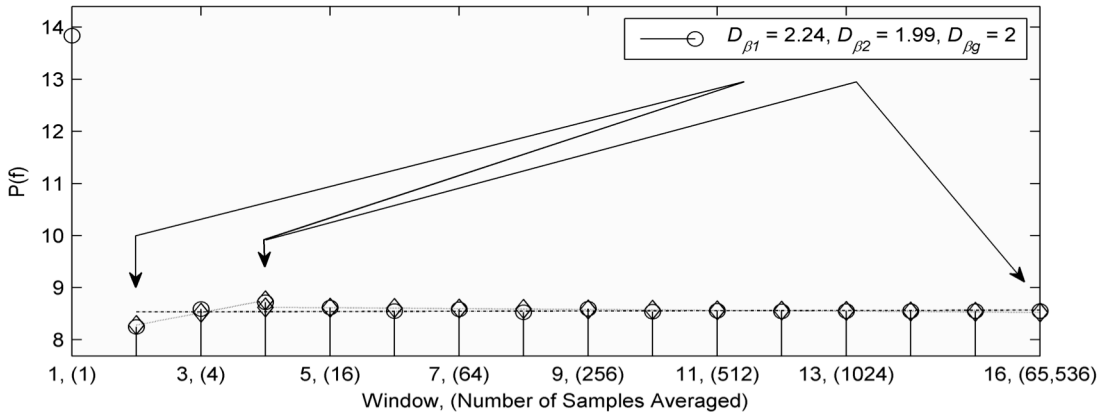


Fig. 7.30. Spectral fractal dimension (D_β) of the cryptosystem based on chaos. Two different fractal sections, $D_{\beta 1} = 2.24$ and $D_{\beta 2} = 1.99$, are shown in the plot. The general SFD is $D_{\beta g} = 2$.

based on RBN exhibits more complexity for smaller windows than $n_w = 2^8$ when compared with ElGamal. The cryptosystem based on chaos shown in Fig. 7.30 has an SFD section that is very close to 2 for a window size even smaller than all the other cryptosystems. This window size is as small as $n_w = 2^3$ samples. Once again, a cryptosystem having a complexity value close to 2 implies that it has a high scrambling degree, which is a desired feature when protecting a

plaintext.

The multifractal analysis using the SFD reveals that the chaos-based cryptosystem is a monofractal whereas the RSA16, ElGamal, and the RBN cryptosystems have a multifractal behaviour with the selected encryption experiments. The monofractality in the chaos-based cryptosystem is advantageous against polyscale-based cryptanalysis because of the non-existence of patterns. This lack of patterns in the chaos-based cryptosystem makes very difficult the use of polyscale cryptanalysis when attempting to break the protected data. The multifractal behaviour caused by the slope changes in the SFD log-log plot found in the counterparts of the chaos-based cryptosystem provides hooks that could be used to break them. Thus, making them weak. The RSA16 ciphertext in Fig. 7.27 has two different slopes in the SFD log-log plot (i) $D_\beta = 2.35$ from $n_w = 2^1$ to $n_w = 2^4$, and (ii) $D_\beta = 1.93$ from $n_w = 2^5$ to $n_w = 2^{10}$. The ciphertext obtained through ElGamal in Fig. 7.28 has three different slopes in the SFD log-log plot (i) $D_\beta = 1.7$ from $n_w = 2^1$ to $n_w = 2^4$, (ii) $D_\beta = 1.6$ from $n_w = 2^5$ to $n_w = 2^7$, and (iii) $D_\beta = 1.98$ from $n_w = 2^7$ to $n_w = 2^{16}$. The RBN ciphertext in Fig 7.29 has two different slopes in the SFD log-log plot (i) $D_\beta = 1.91$ from $n_w = 2^1$ to $n_w = 2^7$, and (ii) $D_\beta = 1.99$ from $n_w = 2^7$ to $n_w = 2^{16}$. Furthermore, Fig. 7.30 shows that the cryptosystem based on chaos has better complexity features than the RSA16, ElGamal, and even the cryptosystem based on a RBN.

7.7 Summary

Four different experiments have been presented across this chapter. The experiments have been implemented considering the constraints stated in the previous chapter. By this, the cryptosystem is capable of protecting, in a similar degree, data from different sources. This

feature has been demonstrated using: a text string, a sound wave, a grayscale picture, and a color picture. The examination of these experiments by relying on the CTs histograms sustain that the cryptosystem is also capable of resisting statistical cryptanalysis. The visual comparison of the ciphertexts verifies that the cryptosystem resembles a one-time pad. This feature because of the similitude with the ciphertext obtained through the data stream produced by RBN. The cryptosystem is also compared with alternative encryption schemes like RSA, ElGamal, and a RBN stream. The comparison considers the framework where a grayscale image is encrypted. This framework allows for (i) identification of patterns in the ciphertext, (ii) speed of encryption/decryption, (iii) FSS10 analysis of ciphertext, and (iv) analysis of the ciphertext complexity by using the SFD. The discussion of the experimental results sustain that this cryptosystem has a strong potential to protect data securely. The importance of the stationarity map and surrogate data has also been stressed. This cryptosystem avoids the classical attacks against chaos-based cryptosystems. The cryptosystem has been compared with alternative encryption schemes as RSA, ElGamal, and a cryptosystem based on RBN. Both the stationarity analysis FSS10 and SFD sustain that the chaos-based cryptosystem has the best performance. As a grand finale, the succeeding chapter provides the conclusions achieved throughout this research.

CHAPTER VIII

CONCLUSIONS

8.1 Main Findings

This thesis presents an efficient new cryptosystem based on modular continuous-interval cellular automata (CCA). The efficiency of the CCA is the basis to generate high-speed chaotic sequences used in the new modular dynamical cryptosystem. Distinct modules containing CCA have been integrated to generate short chaotic sequences that can be combined to form a chaotically-mixed sequence with higher variance. Since the cryptosystem is applicable to distinct sources of data (*e.g.*, text, sound, grayscale images, and color images) it can be used in different applications that require storing and/or transmitting information in a secure way. A CCA capable of exhibiting a chaotic regime is the core of the proposed cryptosystem. The dynamical system in which this cryptosystem is based is introduced in this thesis for the first time. This dynamical system, being the smallest reported on the literature, is capable of exhibiting chaos. The implementation of this cryptosystem has the advantages of being fast and flexible. These advantages can allow it to overcome the difficulty of implementing cryptosystems based on *quantum cryptography* (QC), which is often considered as a technological challenge. Shannon's observation mentioned in the introduction about good mixing transformations made by non-commuting operations is transcendental and extremely important through the whole process of the cryptosystem operation.

The encryption/decryption processes provided by the new cryptosystem are of value to both civilian and military industries. This is because of the very large number of keys provided by the

variation of initial conditions in the chaotic sequences. Most of the approaches presented in the available literature consider CA with limited capabilities, limited number of keys, and a unique module that describes the entire cryptosystem, making it vulnerable or predictable.

For the first time, a method of precise determination of the window size in which a signal is stationary in the FSS10 sense has been introduced. This testing method considers a stronger demand than WSS stationarity. A graphical analysis is provided by this method known as *stationarity map* that allows evaluating if the first 10 moments of a time series are indeed time independent. This method has been verified using a known PRNG having a uniform distribution and when applied to the cryptosystem it has been found that the cryptosystem is time invariant for a window size as small as $n_w = 2^9$. The framework provided by the FSS10 is consistent with the classical understanding of the WSS and SSS and the number of statistical moments could increase or decrease (*e.g.*, FSS7, if seven statistical moments are considered). This is important to achieve a higher degree of security for the cryptosystem. The FSS10 method developed has been verified to work properly even when using a small number of elements in a window (*i.e.*, a few thousand).

The degree of complexity of a dynamical system based on CA is measured for the first time through polyscale methods. The methods used are the VFDT and the SFD, which provide a quantitative perspective of the amount of chaos in a time series. The implementations of both methods are verified using a PRNG to assess that they can detect it as a space-filling function. These methods further determine that the cryptosystem is indeed of fractal nature, achieving the highest possible complexity for a space-filling function, as shown by the $D_\sigma \cong 2$ and the $\beta \sim 0$ corresponding to a $D_\beta \cong 2$. These methods provide an important contribution as alternatives to

study, analyze, and compare dynamical systems. This cryptosystem has white noise like properties (*e.g.*, RNG with uniform distribution).

The reduction of the *Spanish moss effect* in the PSD is addressed in a dyadic form, and is modeled by the power spectrum from the piano. This experiment allows understanding of: (i) the frequency doubling behaviour in the logarithmic scale of the frequency domain and (ii) why averaging over equally spaced segments in the frequency support helps in eliminating the *Spanish moss effect*. Through this experiment it is possible to determine the power spectrum of a piano as $\beta \sim 12\log_{10}\left(\sqrt[12]{2}\right)$, which keeps the constant ratio of $\sqrt[12]{2}$ over adjacent notes. This is the first reported experimental verification concluding that equal spaced averaging in the frequency support reveals the true envelope of any time series. This realization then is useful in determining properly an SFD.

The methods for the chaotic mixing of short chaotic sequences and the concealment of a chaotic attractor through surrogate data are presented for the first time by this thesis. Both methods contribute to reducing the classical flaws that chaos-based cryptosystems could be vulnerable to: (i) identification of the chaotic system in the time domain, (ii) reconstructing the chaotic attractor by monitoring the system state variables, and (iii) searching the system synchronization parameters. The cryptosystem resilience to these attacks constitutes its most outstanding advantage. The inherent strong *avalanche effect* in this cryptosystem is one of its most important characteristics. This is achieved through the very nature of a chaotic sequence (*i.e.*, dependence on initial conditions) and is amplified by means of the chaotic mixing. Also the *one-time pad* profile used on its development is remarkable to obtain *perfect secrecy* on any of the PTs enciphered. Chaotic mixing allows increasing the complexity of a bit stream as shown

by the VFDT analysis. The chaotic attractor concealment by surrogate data is an extra protective layer. This protection reinforces the inherent uniform distribution of the ciphertexts obtained through this chaos-based cryptosystem. The surrogate data generation method has been designed to have no data loss during the encryption/decryption processes. The extra protection layer provided by concealing the *multi-needle attractor* contributes to the features that make this cryptosystem more secure than other chaos-based cryptosystems. This strong feature in the cryptosystem comes from the alternative data provided by the surrogate data, which cannot be linked directly to the original data source. Therefore, the chaotic process used in the cryptosystem cannot be directly monitored, thus making it extremely hard to break. Applying surrogate data to the cryptosystem provides no alteration to its strong FSS10 characteristics because its statistical moments are not modified. The concealment of a chaotic attractor through the surrogate data generation method is verified using the Hénon attractor and further tested in the cryptosystem. Both cases visually determine that the chaotic attractor is indeed concealed, thus providing stealthy capabilities to the cryptosystem.

Based on the experimental work presented and the positive results achieved through the FSS10, VFD, SFD, and surrogate data, confirm that the cryptosystem offers a high-level of security. These methods indicate thus far that, there is no other chaos-based cryptosystem as strong as the one that has been enhanced and tested here. The research discussed in this thesis reveals an advance for possible applications computer security.

This cryptosystem has several advantages: (i) it eliminates known classical attacks, (ii) establishes a fence for the known attacks provided that the encryption sequence is at least the same length as the PT, and that synchronization is not used, (iii) provides an opportunity to resist

attacks not reported so far through the use of surrogates, and (iv) attempts to establish a possible new research direction on how chaotic systems should be used and implemented for cryptographic applications. It is also important to note that the proposed cryptosystem can be used for symmetric and asymmetric applications.

This CCA-based cryptosystem is the first implementation of a cryptosystem exploiting chaos phenomena. The chaos phenomena are created inside a computer through the *multi-needle* chaotic attractor and tightly harnessed to achieve data protection. The multi-needle chaotic attractor, introduced for the first time in this thesis, is the smallest known chaotic attractor in literature. By using a limited hardware experimental platform (a laptop Fujitsu LifeBook A Series with an Intel Centrino Duo microprocessor and 1 GB in RAM runs Matlab 2012 in Windows 7), it has been shown that the encryption sequences obtained and used for data protection are generated much faster then other cryptosystems using DEs. This difference varies from seconds to weeks of processing time respectively. The uniform histograms of the experimental ciphertexts show that the cryptosystem keeps its properties, consistent while protecting data from distinct sources. The implication of this conclusion is that the cryptosystem is capable of resisting statistical cryptanalysis. The mpmf comparison of RSA, ElGamal, RBN, and the chaos-based cryptosystems shows that the CCA-based cryptosystem has the most uniform distribution. This feature is highly advantageous when designing cryptosystems capable of resisting statistical cryptanalysis.

The CCA-based cryptosystem is also resistant to a new form of cryptanalysis based on polyscale signal processing, as shown by the analysis of the ciphertexts by the SFD. From the SFD analysis it is seen that the cryptosystem is a monofractal for a window as small as $n_w = 2^3$.

This test surpasses by far the other cryptosystems ($n_w = 2^7$ for the ElGamal and the RBN based cryptosystem) because it provides no *hooks* that could weaken it. All the counterparts analyzed (*i.e.*, RSA, ElGamal, and the RBN based cryptosystem) exhibit multifractal nature and are less complex than this chaos-based cryptosystem. These multifractal cryptosystems are susceptible to attacks that could succeed in revealing the plaintexts protected by them.

A high-speed implementation of a cryptosystem is provided. This implementation surpasses the RSA and ElGamal implementations by four and two orders of magnitude respectively when encrypting. The RSA and ElGamal are slower than the chaos-based cryptosystem by two and one order of magnitude as seen in the decryption experiments performed. The chaos-based cryptosystem performs also faster than the RBN based cryptosystem because the bit streams are produced locally, and do not depend of an external process (*i.e.*, quantum cryptography). This compact design of the chaos-based cryptosystem also limits the flaws that could be exploited by attackers. This cryptosystem is flexible to perform data protection either in symmetric or asymmetric cryptographic schemes.

8.2 Answers to the Research Questions Posed in this Thesis

It is demonstrated that the simple cryptosystem based on CCA exhibits complex behaviour to secure data. This complex behaviour has been tested to be comparable to the characteristics of white noise by the VFDT and the SFD.

It is shown that this cryptosystem is capable of protecting data from distinct sources like text, sound, grayscale images, and color images, quite uniformly, leaving no patterns in the ciphertexts.

The stationarity test using the FSS10 shows that the cryptosystem has time invariance for a

window size small as $n_w = 2^9$.

The cryptosystem is based on a CCA that exhibits chaos-like behaviour that is comparable to chaotic systems based on DEs.

This cryptosystem overcomes vulnerabilities like identification of a chaotic system in the time domain, model reconstruction by monitoring state variables, and synchronization search. This achievement is based on (i) using surrogate data to conceal the multi-needle attractor and (ii) avoiding the use of synchronization between the encrypter/decrypter systems. Removing these vulnerabilities strengthens the cryptosystem against cryptanalysis.

The research described in this thesis, together with the experiments performed, and analysis methods used demonstrate that

> *this cellular-automata-based cryptosystem is capable of exhibiting chaotic behaviour that is FSS10 stationary. This cryptosystem does not include the classical vulnerabilities present in chaos-based cryptosystems and, it is capable of resisting statistical and polyscale cryptanalysis attacks.*

In the context of the work done, this is the formal answer to the thesis question provided in the introduction chapter. Saving the best for the last, the dynamical system in which the cryptosystem presented here is constructed requires a single equation. It is remarkable that this extremely compact representation of a dynamical system exhibits chaotic behaviour. This dynamical system constitutes the smallest one that has been published in the literature so far.

8.3 Contributions

(a) A cryptosystem capable of protecting data robustly. It has been demonstrated that this cryptosystem is capable of creating ciphertexts that are (i) FSS10 stationary and (ii) highly complex as sustained by the VFDT and SFD analysis. It is expected that this cryptosystem could resist robustly statistical and polyscale cryptanalysis attacks.

(b) A design methodology. Through the development of the MMDC, a new methodology to design chaos-based cryptosystems is shown. This methodology incorporates the interaction of different dynamical systems running in different modules.

(c) The simplest known chaotic attractor. The Rössler and Hénon chaotic attractors have been considered the most compact dynamical systems until now. The first one a dynamical system (flow) using a set of three DEs while the last one is a dynamical system (map) using a set of two discrete equations. The *multi-needle attractor* presented in this thesis is the simplest known dynamical system capable of exhibiting chaos phenomena using only one discrete equation. The route to chaos of the *multi-needle attractor* by selecting values for its single parameter is also presented.

(d) Chaotic mixing. The fusion of different sequences through chaos phenomena is demonstrated. This process allows increasing the complexity of the resulting sequence. Through the development of the MMDC, a new methodology to design chaos-based cryptosystems is shown.

(e) Chaotic attractors concealed through surrogate data. Chaotic attractors are made unintelligible by using surrogate data. This provides a stealth protection layer to the MMDC cryptosystem through against cryptanalysis attacks.

(f) Classical vulnerabilities of chaos-based cryptosystems are removed. Cryptosystems based on chaos have been considered weak against (i) identification of the chaotic system in the time domain, (ii) reconstructing the chaotic attractor by monitoring the system state variables, and (iii) searching the system synchronization parameters. The cryptosystem introduced in this thesis removes all these vulnerabilities by chaotic mixing and the chaotic attractor concealment.

(g) The FSS10 method. The FSS10 method through the stationarity map can clearly identify if a data stream or a signal is stationary. This method uses a middle range stationarity measure that does not fall into the extremes of WSS and SSS. However, there are three constrains when implementing FSS10: (i) the growing error, (ii) the computational power required, and (iii) the vanishing moments. These constrains are challenging when considering more than 10 moments in an implementation of *finite sense stationarity*.

(h) The stationarity map. This method reveals precisely the window size in which a signal is FSS10 stationary.

8.4 Novelty in the Thesis

(a) The simplest known chaotic attractor. The *multi-needle attractor* is the simplest known dynamical system capable of exhibiting chaos phenomena using only one discrete equation having a single parameter.

(b) The concept of chaotic mixing. The idea of using a chaotic attractor as way of scrambling sequences that are already chaotic has been demonstrated in this thesis.

(c) Chaotic attractor concealment. The capability of surrogate data to conceal a chaotic attractor and make it visually look like random noise has been also conceived and

implemented.

(d) Polyscale methods to measure the degree of complexity of a dynamical system. There is no reported literature considering a method to measure the amount of chaos that a dynamical system based on CA could exhibit. The polyscale methods addressed here bring a quantitative basis for comparison among dynamical systems. These methods are capable of establishing a degree for the complexity of dynamical systems.

(e) The FSS10 method. There is no stationarity method known that considers higher order moments to the ones considered by the WSS. The FSS10 method through the stationarity map can clearly identify if a data stream or a time series is stationary. A middle range stationarity method that does not fall into the extremes of WSS and SSS has been introduced.

(f) The stationarity map. An outstanding method that reveal precisely the window size in which a time series is FSS10 stationary. The significance of *the stationarity map* in this research is quite substantial, given the fact that a robust method to determine the window size in which a time series is stationary has not been reported thus far in the literature.

REFERENCES

[AlBa11] Tansu Alpcan and Tamer Başar, *Network Security A Decision and Game-Theoretic Approach*. Cambridge, NY: Cambridge University Press, 2011.

[Angh11] Petre Anghelescu, "Encryption algorithm using programmable cellular automata," in *2011 World Congress on Internet Security (WorldCIS)*, pp. 233-239, 21-23 February 2011.

[AnIS08] Petre Anghelescu, Silviu Ionita and Emil Sofron, "FPGA implementation of hybrid additive programmable cellular automata encryption algorithm," in *Proc. International Conference on Hybrid Intelligent Systems, 2008. HIS '08*, pp. 96-101, September 2008.

[AnSI07] Petre Anghelescu, Emil Sofron and Silviu Ionita, "VLSI implementation of high-speed cellular automata encryption algorithm," in *Proc. International Semiconductor Conference, 2007. CAS 2007*, vol. 2, pp. 509-512, October-September 2007.

[Assc06] Gilles Van Assche, *Quantum Cryptography and Secret-Key Distilation*. New York, NY: Cambridge University Press, 2006, 280 pp.

[ASRI08] Petre Anghelescu, Emil Sofron, Cristian-I. Rîncu and Vasile-G. Iana, "Programmable cellular automata based encryption algorithm," in *Proc. International Semiconductor Conference, 2008. CAS 2008*, vol. 2, pp. 351-354, October 2008.

[ATHB11] W.N.A.W. Ali, A.H.M. Taib, N.M. Hussin, R. Budiarto, and J. Othman, "Distributed security policy for IPv6 deployment," in *3rd International Symposium & Exhibition Sustainable Energy & Environment (ISESEE), 2011*, pp. 120-124, 1-3 June 2011.

{doi: 10.1109/ISESEE.2011.5977081}

[BeBr84] Charles H. Bennett and Gilles Brassard, "Quantum cryptography: Public key distribution and coin tossing," in *Proc. IEEE 1984 International Conference on Computers, Systems and Signal Processing* (Bangalore, India), pp. 175-179, December 1984.

[Bloo06] Peter Bloomfield, *Fourier Analysis of Time Series*. Danvers, MA: Wiley-Interscience, 2006 (2nd ed.), 275 pp.

[BoBC09] Apaporn Boonyarattaphan, Yan Bai and Sam Chung, "A security framework for e-Health service authentication and e-Health data transmission," in *Communications and Information Technology, 2009. ISCIT 2009. 9th International Symposium*, pp. 1213-1218, September 2009.

[Bows82] Bruce Bosworth, *Codes, Ciphers, and Computers: An Introduction to Information Security*. Rochelle Park, NJ: Hayden, 1982, 259 pp.

[ChAb11] T. M. Chen and S. Abu-Nimeh, "Lessons from Stuxnet," in *Computer*, vol. 44, no. 4, pp. 91-93, April 2011.
{doi: 10.1109/MC.2011.115}

[Cons11] L. Constantine, "Crossing the line: Terrorism in cyberspace and targets in real-space," in *2011 International Conference on Cyberworlds (CW)*, pp. 1-4, 4-6 October 2011.
{doi: 10.1109/CW.2011.49}

[Coop81] Paul Cooper, *Perspectives in Music: An Historical-Analytical Approach*. New York, NY: Harper & Row, 1981 (2nd ed.), 564 pp.

[CuOp93] Kevin M. Cuomo and Alan V. Oppenheim, "Circuit implementation of synchronized

chaos with applications to communications," in *The American Physical Society Journal*, pp. 65-68, 1993.

[Daub92] Ingrid Daubechies, Ten Lectures on Wavelets. Philadelphia, PA: SIAM, 1992, 357 pp.

[DeGu10] Dessalegn Y. Melesse, Abba B. Gumel, "Global asymptotic properties of an SEIRS model with multiple infectious stages," in *Trans. Journal of Mathematical Analysis and Applications - J MATH ANAL APPL*, vol. 366, no. 1, pp. 202-217, 2010. doi: 10.1016/j.jmaa.2009.12.041

[Deva92] Robert L. Devaney, *A First Course in Chaotic Dynamical Systems: Theory and Experiment*. Reading, MA: Addison-Wesley, 1992, 302 pp.

[DiHe76] Whitfield Diffie and Martin E. Hellman, "New directions in cryptography," in *IEEE Tran. Information Theory*, vol. 22, no. 6, pp. 644-654, November 1976.

[DiKl10] Yun Ding and Karsten Klein, "Model-driven application-level encryption for the privacy of e-Health data," in *Proc. Availability, Reliability, and Security, 2010. ARES '10 International Conference*, pp. 341-346, February 2010.

[Dola01] Kevin T. Dolan, "Surrogate for nonlinear time series analysis," in *Phys. Rev. Let. statistical, nonlinear, and soft matter physics, The American Physical Society*, vol. 64 pp. 0461281-0461286, 2001.

[EXKL03] I. Elhajj, Xi Ning, Wai Keung Fung, Yun-Hui Liu, Y. Hasegawa, and T. Fukuda, "Supermedia-enhanced Internet-based telerobotics," in *Proc. of the IEEE*, vol. 91, no. 3, pp. 396- 421, March 2003. {doi: 10.1109/JPROC.2003.809203}

[Ferg10] Niels Ferguson, *Cryptography Engineering: Design Principles and Practical*

Applications. New York, NY: Wiley, 2010, 384 pp.

[Gain56] Helen Fouché Gaines, *Cryptanalysis: A Study of Ciphers and Their Solution*. New York, NY: Dover, 1956, 256 pp. (Reprinted 2000.)

[Gait78] Jason Gait, "Encryption standard: Validating hardware techniques," in *Dimensions*, vol. 62, no. 7/8, pp. 22-24, July-August 1978.

[Gent09] Craig Gentry, "Fully homomorphic encryption using ideal lattices," in *Proc. of the 41st Annual ACM Symposium on Theory of Computing, STOC'09* (Bethesda, MD), 2009. {ISBN: 978-1-60558-506-2}

[Gior10] Nicholas J. Giordano, *The Physics of the Piano*. New York, NY: Oxford University Press, 2010, 170 pp.

[GlDu91] Neal Glover and Trent Dudley, *Practical Error Correction Design for Engineers*. Broomfield, CO: Cirrus Logic-Colorado, 1991 (2nd ed.

[Gran11] Clarence Grant Hamilton, *Sound and Its Relation to Music*. Toronto, Canada: University of Toronto Libraries, 2011, 168 pp.

[Gröc01] Karlheinz Gröchenig, *Foundations of Time-Frequency Analysis (Applied Numerical Harmonic Analysis)*. Boston, MA: Birkhäuser, 2001.

[Haah12] Mads Haahr, True Random Number Service. [Online]. Available: www.random.org

[HaSo96] Gary D. Hachtel and Fabio Somenzi, *Logic Synthesis and Verification Algorithms*. New York, NY: Springer Science & Business Media, 1996, 564 pp.

[Helm54] Hermann Helmholtz, *On the Sensations of Tone*. New York, NY: Dover Publications, 1954,608 pp.

[Héno76] Michel Hénon, "A two-dimensional mapping with a strange attractor," in *Comm.*

Math. Phys., vol. 50, pp. 69-77, 1976.

[HiHS83] R. Hinden, J. Haverty, and A. Sheltzer, "The DARPA Internet: Interconnecting heterogeneous computer networks with gateways," in *Computer*, vol. 16, no. 9, pp. 38-48, September 1983.
{doi: 10.1109/MC.1983.1654494}

[HMHS12] Johan Heyszl, Stefan Mongard, Benedikt Heinz, Frederic Stumpf, and Georg Sigl, "Localized electromagnetic analysis of cryptographic implementations," in *Cryptology – CT–RSA 2012. Orr Dunkelman (Ed.)*. Berlin, Germany: Springer, 2012, pp. 231-244.

[HuKl09] Ji Hu and Andreas Klein, "A benchmark of transparent data encryption for migration of web applications in the cloud," in *Proc. IEEE 8th International Conference on Dependable, Autonomic and Secure Computing, 2009. DASC '09.*, pp. 735-740, December 2009.

[Kahn67] David Kahn, *The Codebreakers: The Story of Secret Writing*. New York, NY: The Macmillan Company, 1967, 475 pp. (Reprinted 1996.)

[KaLi07] Jonathan Katz and Yehuda Lindell, *Introduction to Modern Cryptography: Principles and Protocols*. Virginia Beach, VA: Chapman & Hall/CRC, 2007, 552 pp.

[Karn11] S. Karnouskos, "Stuxnet worm impact on industrial cyber-physical system security," in *IECON 2011 - 37th Annual Conference on IEEE Industrial Electronics Society*, pp. 4490-4494, 7-10 November 2011.
{doi: 10.1109/IECON.2011.6120048}

[KaOP2010] Timo Kasper, David Oswald, and Christof Paar, "A versatile framework for implementation attacks on cryptographic RFIDs and embedded devices," in *Trans. on Computational Science X. Marina Gavrilova, C. Tan, and Edward Moreno (Eds.)*. Berlin, Germany: Springer, 2010, pp. 100-130.
{doi: 10.1007/978-3-642-17499-5_5}

[KaSc04] Holger Kantz and Thomas Schreiber, *Nonlinear Time Series Analysis*. New York, NY: Cambridge University Press, 2004 (2nd ed.), 388 pp.

[KCCP06] Witold Kinsner, Vincent Cheung, Kevin Cannons, Joe Pear, and Toby Martin, "Signal classification through multifractal analysis and complex domain neural networks," in *IEEE Trans. Systems, Man, and Cybernetics*, Part C, vol. 36, no. 2, pp. 196-203, March 2006.

[KeRS00] Michael P. Kennedy, Riccardo Rovatti, and Gianluca Setti, *Chaotic Electronics in Telecommunications*. Boca Raton, FL: CRC Press, 2000, 445 pp.

[KiGr08] Witold Kinsner and Warren Grieder, "Speech segmentation using multifractal measures and amplification of signal features," in *Proc. IEEE 7th Intern. Conf. Cognitive Informatics, ICCI08*, (Palo Alto, CA; 14-16 August 2008) pp. 351-357, 2008.

[KiGr10] Witold Kinsner and Warren Grieder, "Amplification of signal features using variance fractal dimension trajectory," in *Trans. Intern. Journal on Cognitive Informatics and Natural Intelligence*, vol. 4, no. 4, pp. 1-17, October-December 2010.

[Kins91] Witold Kinsner, "Review of data compression methods, including Shannon-Fano, Huffman, arithmetic, Storer, Lempel-Ziv-Welch, fractal, neural network, and wavelet

algorithms," Technical Report DEL91-1, Dept. Electrical and Computer Engineering, University of Manitoba, January 1991, 157 pp.

[Kins03] Witold Kinsner, "Characterizing chaos through Lyapunov metrics," in *IEEE Trans. on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 36, no. 2, pp. 141-151, March 2006.
doi: 10.1109/TSMCC.2006.871132

[Kins07a] Witold Kinsner, "Single-scale measures for randomness and complexity," in *Proc. 6th IEEE International Conference on Cognitive Informatics*, pp. 554-568, 2007.
doi: 10.1109/COGINF.2007.4341936

[Kins07b] Witold Kinsner, "Towards cognitive machines: Multiscale measures and analysis," in *Trans. Intern. J. Cognitive Informatics and Natural Intelligence*, vol. 1, no. 1, pp. 28-38, January-March 2007.

[Kins07c] Witold Kinsner, "A unified approach to fractal dimensions," in *Trans. Intern. J. Cognitive Informatics and Natural Intelligence*, vol. 1, no. 4, pp. 26-46, October-December 2007.

[Kins08] Witold Kinsner, "A unified approach to fractal dimensions," in *Trans. of Journal of Information Technology Research (JITR)*, vol. 1, no. 4, pp. 62-85, 2008.
doi:10.4018/jitr.2008100105

[Kins09] Witold Kinsner, "Challenges in the design of adaptive, intelligent and cognitive systems," in *Trans. Intern. J. Software Science & Computational Intelligence*, vol. 1, no. 3, pp. 16-35, July-September 2009.

[Kins11a] Witold Kinsner, *Fractal and Chaos Engineering: Lecture Notes*. Winnipeg, MB;

University of Manitoba, Jan. 2011, 940 pp.

[Kins11b] Witold Kinsner, "It's time for polyscale analysis and synthesis in cognitive systems," in *Proc. IEEE 10th International Conference on Cognitive Informatics and Cognitive Computing 2011 (ICCI*CC'11)*, (Keynote).

[Kobl95] Neal Koblitz, *A Course in Number Theory and Cryptography*. New York, NY: Springer, 1994 (2nd ed.), 235 pp.

[KJJR11] Paul Kocher, Joshua Jaffe, Benjamin Jun, and Pankaj Rohatgi, "Introduction to differential power analysis," in *Trans. International Journal of Cryptographic Engineering*, vol. 1, no. 1, pp. 5-27, March 2011.
{doi: 10.1007/s13389-011-0006-y}

[Kulc08] Marcin Kulczycki, "Noncontinuous maps and Devaney's chaos," in *Regular -and Chaotic Dynamics*, vol. 13, no. 2, pp. 81-84, 2008.

[JiPe10] Shangzhu Jin and Jun Peng, "Access control for web services based on feedback and decay," in *Proc. IEEE 9th International Conference on Cognitive Informatics (ICCI'10)*, 2010, pp.501-505, 7-9 July 2010.

[Lang11] R. Langner, "Stuxnet: Dissecting a cyberwarfare weapon," in *Security & Privacy, IEEE*, vol. 9, no. 3, pp. 49-51, May-June 2011.
{doi: 10.1109/MSP.2011.67}

[Lore63] Edward N. Lorenz, "Deterministic nonperiodic flow," in *Trans. Journal of the Atmospheric Sciences*, vol. 20, pp.130-141, 1963.

[Mcgr11] Gary McGraw, "Silver Bullet talks with Ralph Langner," in *Security & Privacy, IEEE*, vol. 9, no. 3, pp. 9-14, May-June 2011.

{doi: 10.1109/MSP.2011.66}

[MeSt92] W. Meier and O. Staffelbach, “Analysis of pseudo random sequences generated by cellular automata,” in *Proc. of Advances in Cryptology - EUROCRYPT 91*, no. 547, pp. 186–189, Springer-Verlag, 1992.

[Mene93] Alfred J. Menezes, *Elliptic Curve Public Key Cryptosystems*. New York, NY: Springer, 1993, 144 pp.

[MeOS01] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone, *Handbook of Applied Cryptography*. Boca Raton: FL, CRC Press, 2001, 816 pp.

[MePS11] A. J. B. Mendes, E. H. Paulicena, and W. A. R de Souza, "Quantum cryptography: A direct approach," in *Trans. Salesian Journal on Information Systems* (Revista de Sistemas de Informacao da FSMA), no. 7, pp. 39-48, Jan-Jun 2011.

[MiRe08] D. Micciancio and O. Regev, “Lattice-based cryptography,” in *D.J. Bernstein, J. Buchmann and E. Dahmen (Eds.), Post Quantum Cryptography* (pp. 147-191). Darmstadt, Germany: Springer. {ISBN: 978-3-642-10019-2}

[MMTS07] Kostas Markantonakis, Keith Mayes, Michael Tunstall, Damien Sauveron, Fred Piper, and Nadia Nedjah, “Smart card security,” in *Computational Intelligence in Information Assurance and Security. Ajith Abraham and Luiza Mourelle (Eds.)*. Berlin, Germany: Springer, 2007, pp. 201-233.
{doi: 10.1007/978-3-540-71078-3_8}

[Moon99] Tood K. Moon and Wynn C. Stirling, *Mathematical Methods and Algorithms for Signal Processing*. Upper Saddle River, NJ: Prentice Hall, 1999, 980 pp.

[MoSb10] Claude Moulin and Marco L. Sbodio, “Improving the accessibility and efficiency of

e-Government processes," in *Proc. IEEE 9th International Conference Cognitive Informatics (ICCI'10)*, pp. 603-610, July 2010.

[MRHM12] Aleksandr Matrosov, Eugene Rodionov, David Harley, and Juraj Malcho, "Stuxnet under the microscope", internal distribution document of ESET, Internet: http://go.eset.com/us/resources/white-papers/Stuxnet_Under_the_Microscope.pdf, March 28 2012 [March 28 2012].

[Murr04] J. Murray Barbour, *Tuning and Temperament: A Historical Survey*. Mineola, NY: Courier Dover Publications, 2004, 228 pp.

[NaKC94] S. Nandi, B. K. Kar, and P. Pal Chaudhuri, "Theory and applications of cellular automata in cryptography," in *IEEE Transactions on Computers*, vol. 43, no. 12, pp. 1346-1357, December 1994.
{doi: 10.1109/12.338094}

[Nich98] Randall K. Nichols, *International Computer Society Association Guide to Cryptography*. New York, NY: McGraw-Hill, 1998.

[NBS77] National Bureau of Standards, *Data Encryption Standard*. Washington, DC: NTIS, vol. 46, January 1977.

[NZSC05] Aleksandra Nenadic, Ning Zhang, Qi Shi, and Carole Goble, "DSA-based verifiable and recoverable encryption of signatures and its application in certified e-Goods delivery," in *Proc. IEEE 2005 International Conference on e-Technology, e-Commerce and e-Services*, pp. 94- 99, March-April 2005.

[OHav12] T. O'Haver, *An Introduction to Signal Processing with applications in Chemical Analysis*. College Park, MD; University of Maryland, 2012.

[Ott02] Edward Ott, *Chaos is Dynamical Systems*. Cambridge, NY; Cambridge University Press, 2002 (2nd ed.), 478 pp.

[PaBe09] Christiana Panayiotou and Brandon Bennett, "Critical thinking attitudes for reasoning with points of view," in *Proc. IEEE 8th International Conference on Cognitive Informatics, (ICCI'09)*, pp. 371-377, June 2009.

[PeJS04] Heinz-O. Peitgen, Hartmut Jürgens and Dietmar Saupe, *Chaos and Fractals: New Frontiers of Science*. New York, NY: Springer Science & Business Media, 2004 (2nd ed.), 382 pp.

[Penz96] Walter T. Penzhorn, "Correlation attacks on stream ciphers," in AFRICON, 1996., IEEE AFRICON 4th , vol. 2, pp. 1093-1098 24-27 September 1996. {doi: 10.1109/AFRCON.1996.563052}

[Pete81] W. Wesley Peterson, *Error-Correcting Codes*. Cambridge, MA: MIT Press, 1961, 285 pp.

[PeZL09] Jun Peng, Du Zhang, and Xiaofeng Liao, "Design of a novel image block encryption algorithm based on chaotic systems," in *Proc. IEEE 8th International Conference on Cognitive Informatics, (ICCI'09)*, pp. 215-221, June 2009.

[PfPf11] Charles P. Pfleeger and Shari Lawrence Pfleeger, *Analyzing Computer Security*. Westford, MA: Prentice Hall, 2011, 800 pp.

[RaRo07] Roy Rajarshi, "Synchronization, chaos and consistency," in *Proc. Quantum Electronics and Laser Science Conference, 2007. QELS '07*, pp.1-2, May 2007.

[Rege06] O. Regev, *Lattice-based Cryptography*. In *Cynthia Dwork (Ed.), Advances in Cryptology-CRYPTO 2006*. Berlin, Germany: Springer Vergag, 2006, pp. 131-141.

[Rhee89] Man Young Rhee, *Error Correcting Coding Theory*. New York, NY: McGraw-Hill, 1989, 461 pp.

[RiSA78] Ronald Rivest, Adi Shamir, and Len Adelman, "A method of obtaining digital signatures and public-key cryptosystems". Cambridge, MA: MIT Technical Memo LCS/TM82, April 1977. Also available under the same title from *Communications of the ACM*, vol. 21, no. 2, pp. 120-126, February 1978.

[RoNB08] G. K. Rohde, J. M. Nichols, and F. Bucholtz, "Chaotic signal detection and estimation based on attractor sets: application to secure communications," in *Trans. Chaos-Woodbury- Journal*, vol. 18, no. 1, 2008.

[Rosi99] Michael Rosing, *Implementing Elliptic Curve Cryptography*. Greenwich, CT: Manning Publications, 1998, 338 pp.

[Röss77] Otto Rössler, "Continuous chaos," in *Proc. International Workshop on Synergetics at Schloss Elmau*, 1977.

[Schn95] Bruce Schneier, *Applied Cryptography: Protocols, Algorithms and Source Code in C*. New York, NY: Wiley, 1995 (2nd ed.), 758 pp.

[ShKi10] Dario Schor and Witold Kinsner, "A study of particle swarm optimization for cognitive machines," in *2010 9th IEEE International Conference on Cognitive Informatics (ICCI'10)*, pp. 26-33, 7-9 July 2010.

[Shan49] Claude E. Shannon, "Communication theory of secrecy systems", in *Trans. Bell Systems Technical Journal*, vol. 28, pp. 656-715, 1949.

[ShSh11] Thomas Schreiber and Andreas Schmitz, "Surrogate time series", in *Trans. Physica D*, vol. 142, pp. 346-382, 2011.

[Smit97] Steven W. Smith, *The Scientist and Engineer's Guide to Digital Signal Processing*. San Diego, CA: California Technical Publishing, 1997.

[Sing99] Simon Singh, *The Code Book: The Science of Secrecy From Ancient Egypt to Quantum Cryptography*. New York, NY: Random House, 1999.

[Sink68] Abraham Sinkov, *Elementary Cryptoanalysis: A Mathematical Approach*. New York, NY: Random House, 1968.

[Stin06] Douglas R. Stinson, *Cryptography: Theory and Practice*. Boca Raton, FL: Chapman & Hall/CRC, 2006 (3rd ed.

[Stro00] Steven H. Strogatz, *Non Linear Dynamics and Chaos*. Cambridge, MA: Westview Press, Perseus Books Group, 2000, 260 pp.

[SuHC10] Wen-T. Sung, Yao-C. Hsu, and Kuan-Y. Chen, "Enhance information acquired efficiency for wireless sensors networks via multi-bit decision fusion," in *Proc. IEEE 9th International Conference on Cognitive Informatics (ICCI'10)*, pp. 154-159, July 2010.

[Swen08] Christopher Swenson, *Modern Cryptanalysis: Techniques for Advanced Code Breaking*. New York, NY: Wiley, 2008, 264 pp.

[Take81] Floris Takens, "Detecting strange attractors in turbulence", in *Dynamical Systems and Turbulence, D. A. Rand and L.-S. Young (Eds.)*, Berlin, Germany: Springer-Verlag, 1981, pp. 366–381.

[Tana07] Hidema Tanaka, "Information leakage via electromagnetic emanation and evaluation of tempest countermeasures," in *Information Systems Security. Patrick McDaniel and Shyam Gupta (Eds.)*. Berlin, Germany: Springer, 2007, pp. 167-179.

{doi: 10.1007/978-3-540-77086-2_13}

[TaZh05] Yi Tang and Liankuan Zhang, "Adaptive bucket formation in encrypted databases," in *Proc. IEEE 2005 International Conference in e-Technology, e-Commerce and e-Service, 2005. EEE '05*, pp. 116–119, March-April 2005.

[Tilb03] Henk C. A. van Tilborg, *Fundamentals of Cryptology: A Professional Reference and Interactive Tutorial*, Norwell, MA: Kluwer Academic Publishers, 2003, 500 pp.

[ToMa87] Tommaso Toffoli and Norman Margolus, *Cellular Automata Machines: A New Environment for Modeling*, Cambridge, MA: MIT Press, 1987, 140 pp.

[ToPe01] M. Tomassini and M. Perrenoud, "Cryptography with cellular automata" in *Applied Soft Computing*, no. 1, vol. 2, pp. 151–160, 2001.

[Turc97] Donald L. Turcotte, *Fractals and Chaos in Geology and Geophysics*. Cambridge, UK: Cambridge University Press, 1997 (2nd ed.), 398 pp.

[VaOo89] Scott A. Vanstone and Paul C. van Oorshot, *An Introduction to Error Correcting Codes with Applications*. Boston, MA: Kluwer Academic, 1989, 289 pp.

[Youn39] Robert W. Young, "Terminology for logarithmic frequency units," in *Trans. Journal of the Acoustical Society of America*, vol. 11, pp. 134-139, 1939.

[QuMZ08] Zhiming Qu, Tongbin Ma, and Yunlong Zhang, "Application of parameter modulation in e-Commerce security based on chaotic encryption," in *Electronic Commerce and Security, 2008 International Symposium*, pp. 390-393, August 2008.

[Wash08] Lawrence C. Washington, *Elliptic Curves: Number Theory and Cryptography*. Boca Raton, FL: Chapman and Hall/CRC, 2008 (2nd ed.), 536 pp.

[Wake78] John Wakerley, *Error Correcting Codes, Self-Checking Circuits and Applications*.

New York, NY: North-Holland, 1978, 231 pp.

[WiKP98] A. Witt, J. Kurths, and A. Pikovsky, "Testing stationarity in time series", in *Physical Review E: statistical, nonlinear, and soft matter physics*, vol. 58, pp. 1800-1810, August 1998.

[WLZF01] Xin Wu, Weixian Liu, Lei Zhao, and Jeffrey S. Fu, "Chaotic phase code for radar pulse compression," in *Proc. IEEE National Radar Conf.*, pp. 279-283, 2001.

[Wolf86] Stephen Wolfram, "Cryptography with cellular automata" in *Advances in Cryptology CRYPTO 85. H. C. Williams (Ed.).* Springer, 1986, pp. 429-432.

[Wolf02] Stephen Wolfram, *A New Kind of Science*. Champaign, IL: Wolfram Media, 2002, 1500 pp.

[YaWC97] Tao Yang, Chai Wah Wu, and Leon O. Chua, "Cryptography based on chaotic systems," in *IEEE Trans. Circuits and Systems I: Fundamental Theory and Applications*, vol. 44, no. 5, pp. 469-472, May 1997.

[YCLS09] Yuan Ke-Ya, Chen Jie, Liu Guo-Ping, and Sun Jian, "Design and implementation of data encryption for networked control systems," in *IEEE International Conference on Systems, Man and Cybernetics, 2009. SMC 2009.*, pp. 2105-2109, 11-14 October 2009.
{doi: 10.1109/ICSMC.2009.5346264}

[YiRY09] Weng Yifang, Zheng Rong, and Chen Yi, "A self-synchronous stream cipher based on composite discrete chaos," in *Proc. IEEE 8th International Conference on Cognitive Informatics, (ICCI'09)*, pp. 210-214, June 2009.

[Youn39] Robert W. Young, "Terminology for logarithmic frequency units," in *Trans. Journal*

of the Acoustical Society of America, vol. 11, pp. 134-139, 1939.

[Zhao10] Feng Zhao , "Sensors meet the cloud: Planetary-scale distributed sensing and decision making," in *Proc. IEEE 9th International Conference on Cognitive Informatics (ICCI'10)*, pp. 998, July 2010.

[ZoGu10] Zhong hua Pang and Guoping Liu, "Secure networked control systems under data integrity attacks," in *2010 29th Chinese Control Conference (CCC)*, pp. 5765-5771, 29-31 July 2010.

APPENDIX A

FSS10 VERIFICATION

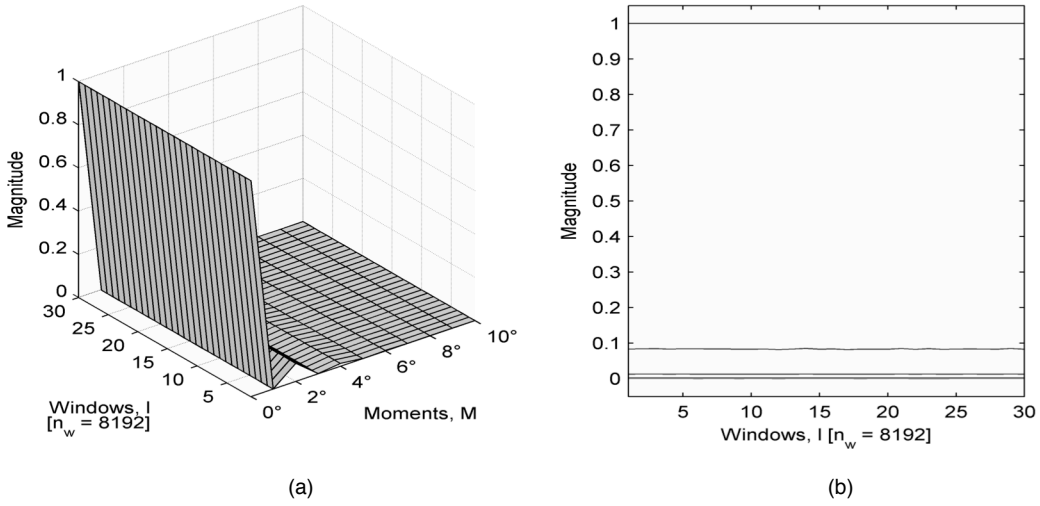


Fig. App.A.1. **(a)** 3D representation of the first ten moments analysis. **(b)** 2D representation of the first ten moments analysis. The window length is $n_w = 2^{13}$ for a time series with uniform distribution in both cases.

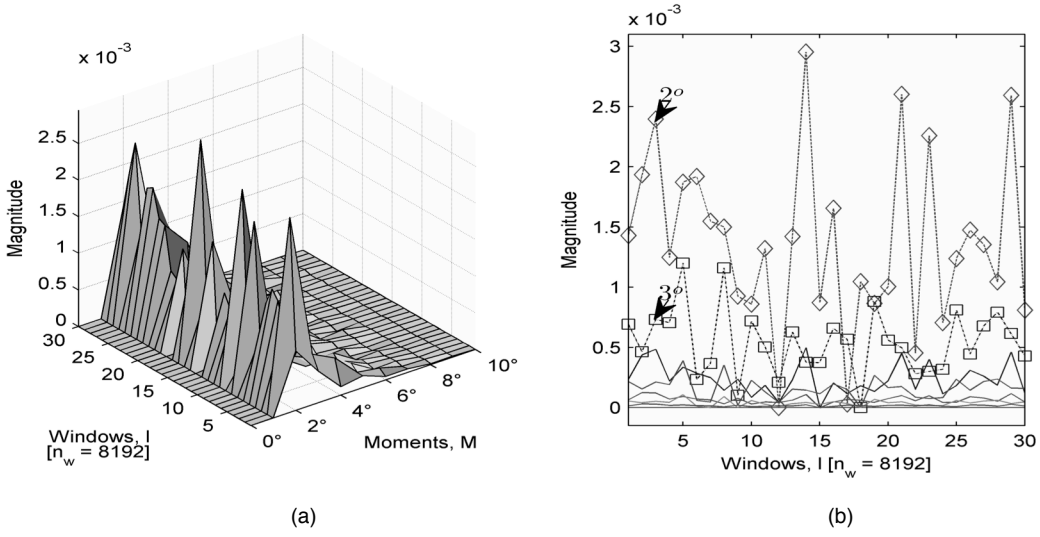


Fig. App.A.2. **(a)** 3D difference representation of the first ten moments analysis. **(b)** 2D difference representation of the first ten moments analysis. The window length is $n_w = 2^{13}$ for a time series with uniform distribution in both cases.

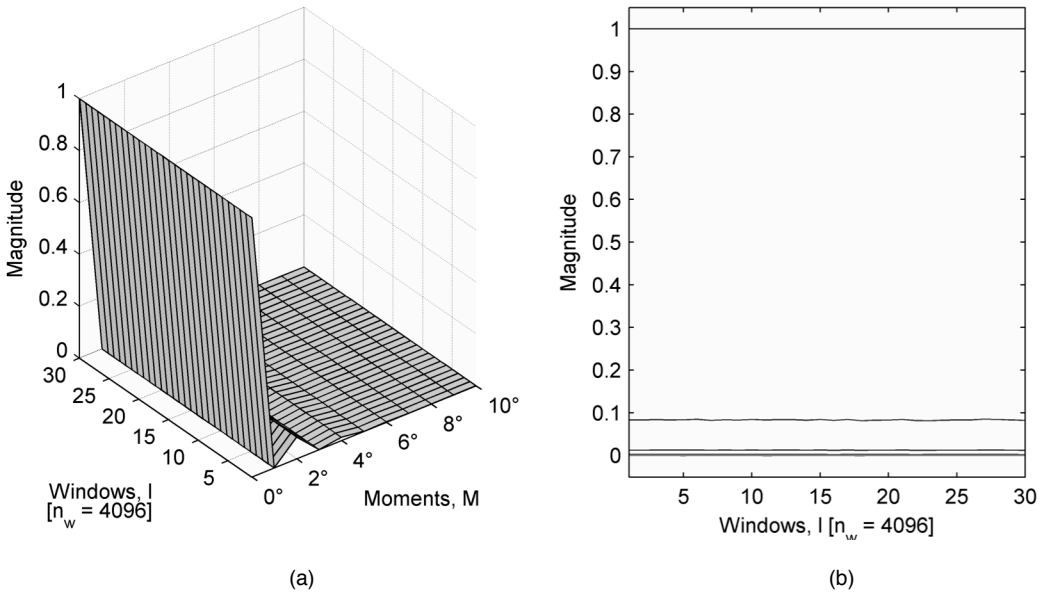


Fig. App.A.3. **(a)** 3D representation of the first ten moments analysis. **(b)** 2D representation of the first ten moments analysis. The window length is $n_w = 2^{12}$ for a time series with uniform distribution in both cases.

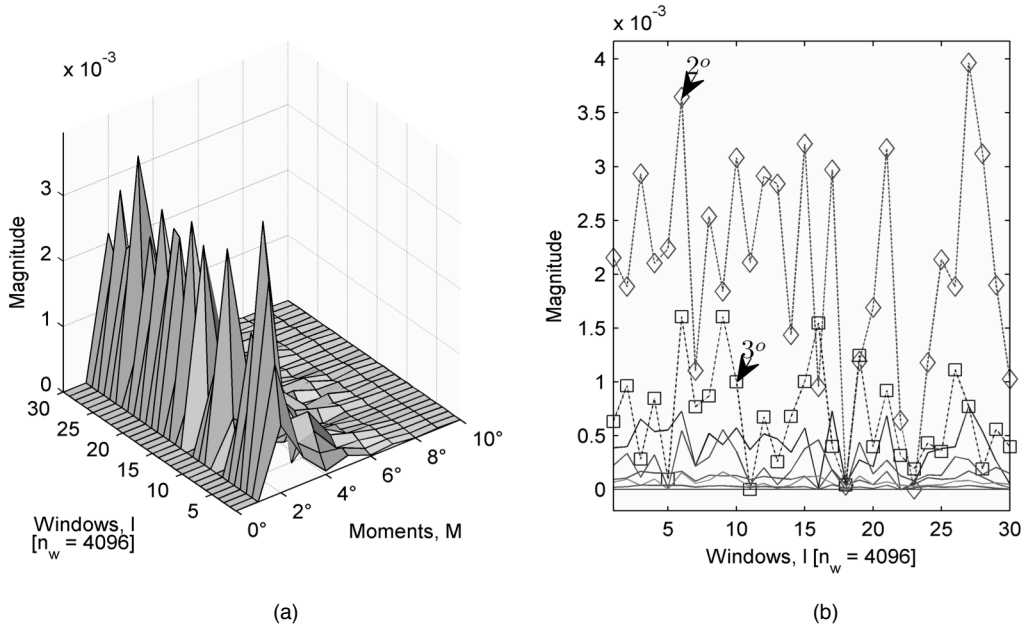


Fig. App.A.4. **(a)** 3D difference representation of the first ten moments analysis. **(b)** 2D difference representation of the first ten moments analysis. The window length is $n_w = 2^{12}$ for a time series with uniform distribution in both cases.

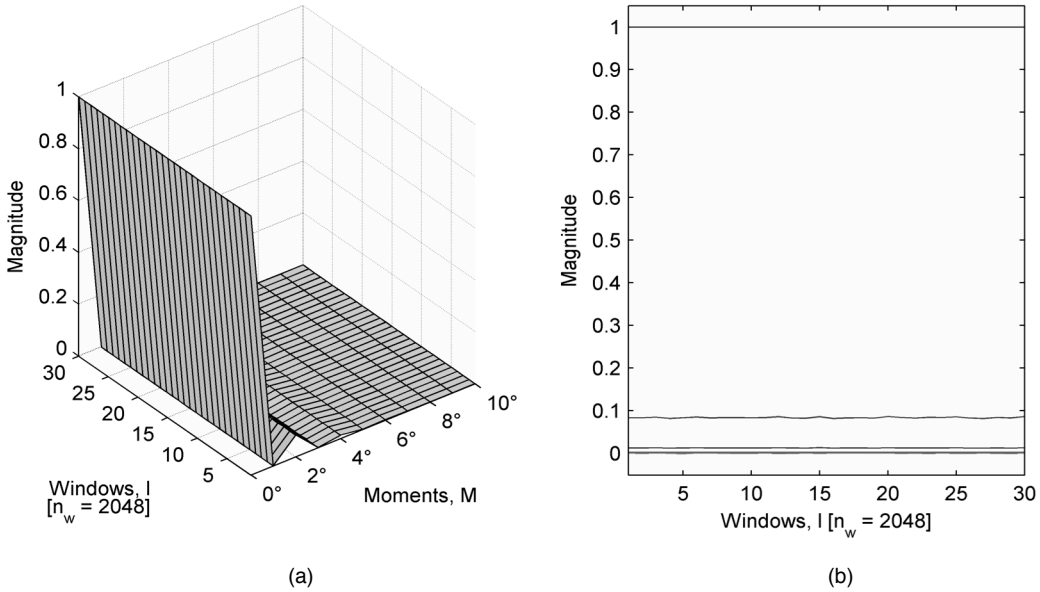


Fig. App.A.5. **(a)** 3D representation of the first ten moments analysis. **(b)** 2D representation of the first ten moments analysis. The window length is $n_w = 2^{11}$ for a time series with uniform distribution in both cases.

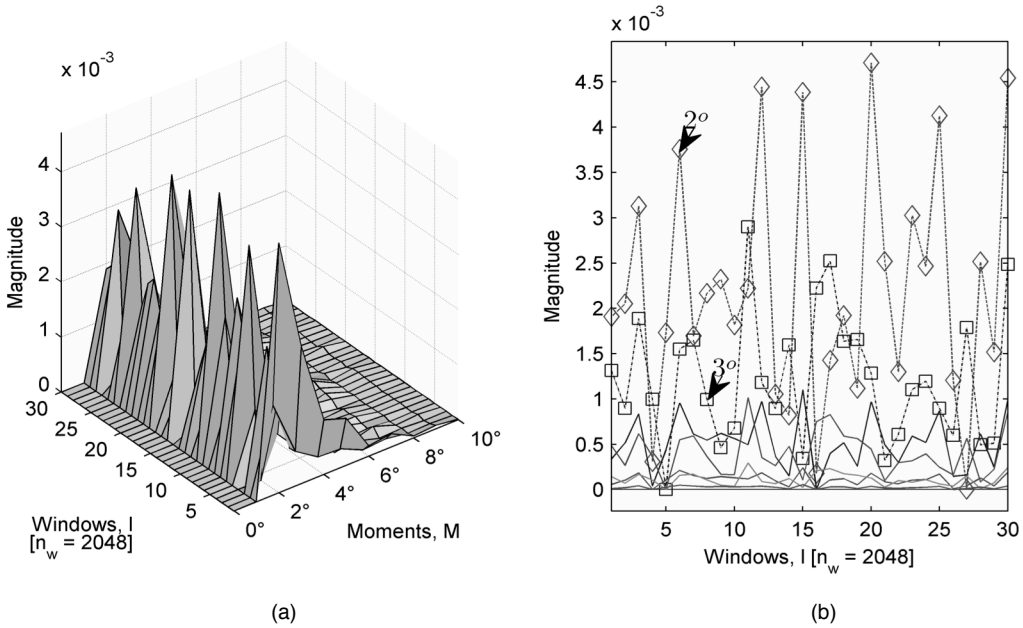


Fig. App.A.6. **(a)** 3D difference representation of the first ten moments analysis. **(b)** 2D difference representation of the first ten moments analysis. The window length is $n_w = 2^{11}$ for a time series with uniform distribution in both cases.

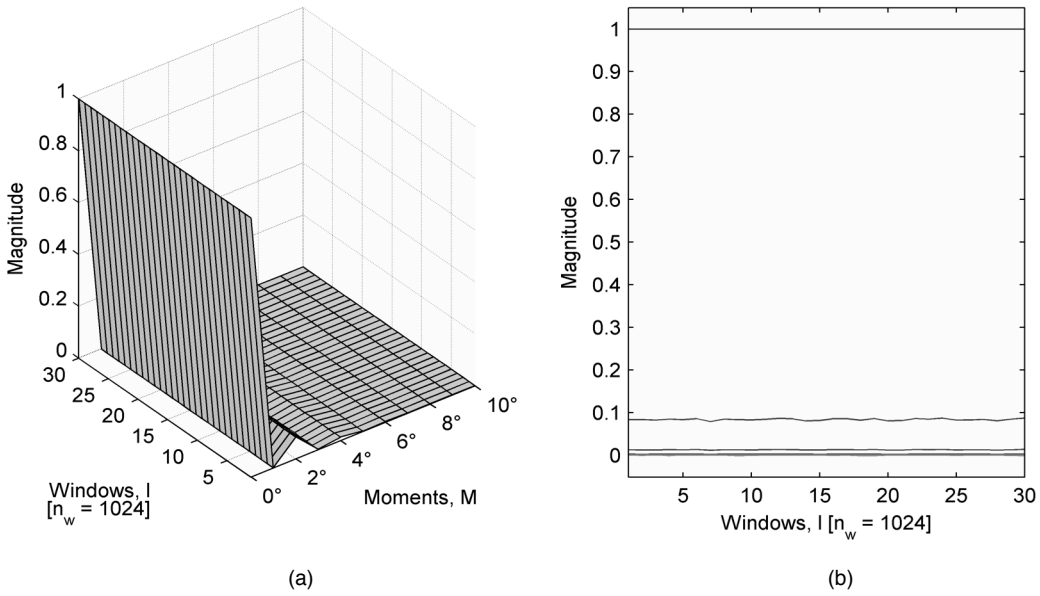


Fig. App.A.7. **(a)** 3D representation of the first ten moments analysis. **(b)** 2D representation of the first ten moments analysis. The window length is $n_w = 2^{10}$ for a time series with uniform distribution in both cases.

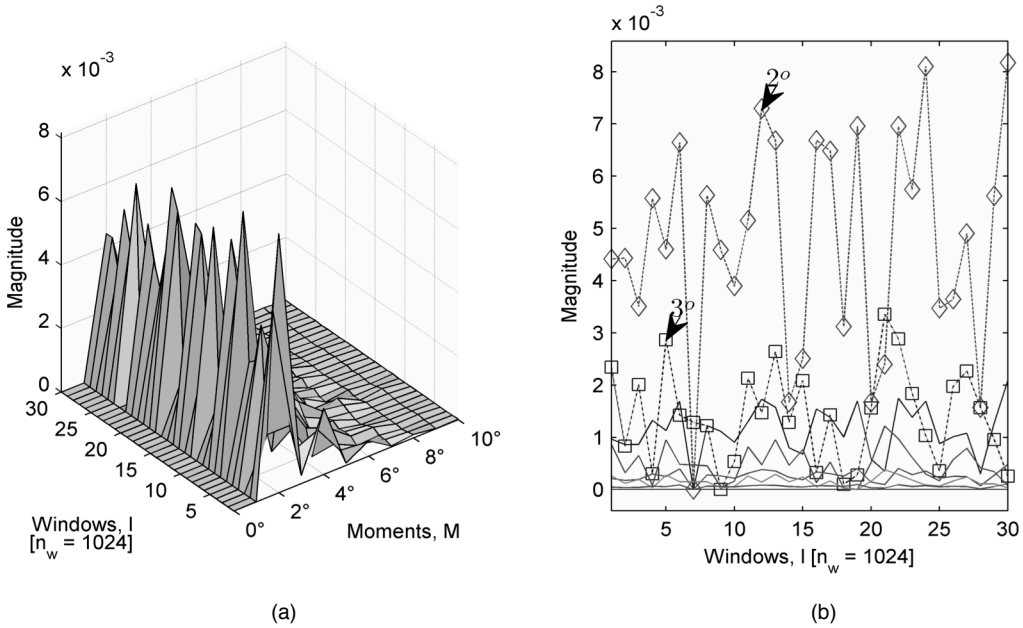


Fig. App.A.8. **(a)** 3D difference representation of the first ten moments analysis. **(b)** 2D difference representation of the first ten moments analysis. The window length is $n_w = 2^{10}$ for a time series with uniform distribution in both cases.

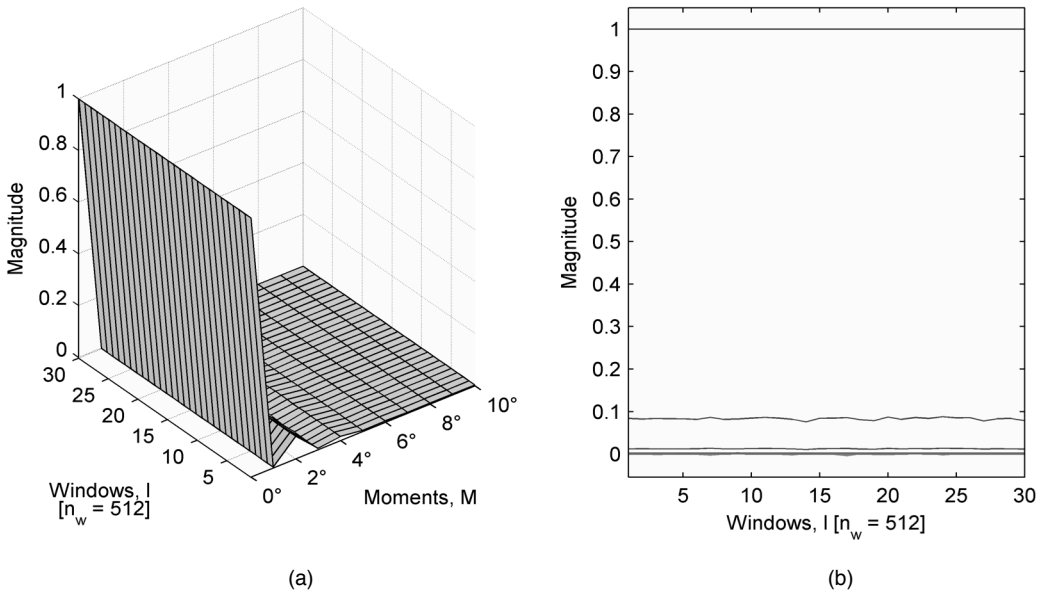


Fig. App.A.9. **(a)** 3D representation of the first ten moments analysis. **(b)** 2D representation of the first ten moments analysis. The window length is $n_w = 2^9$ for a time series with uniform distribution in both cases.

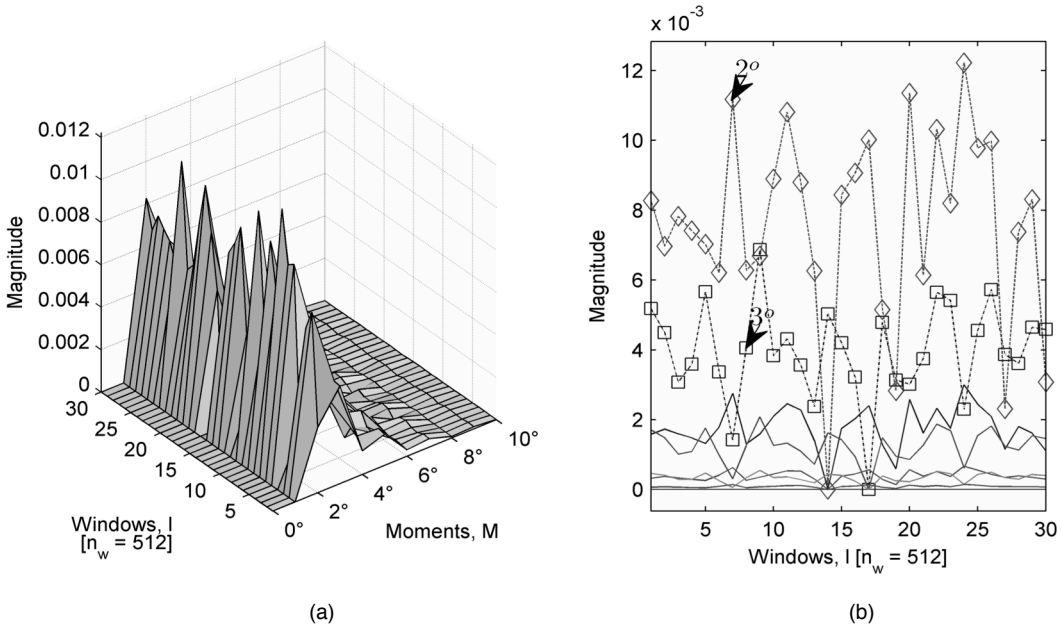


Fig. App.A.10. **(a)** 3D difference representation of the first ten moments analysis. **(b)** 2D difference representation of the first ten moments analysis. The window length is $n_w = 2^9$ for a time series with uniform distribution in both cases.

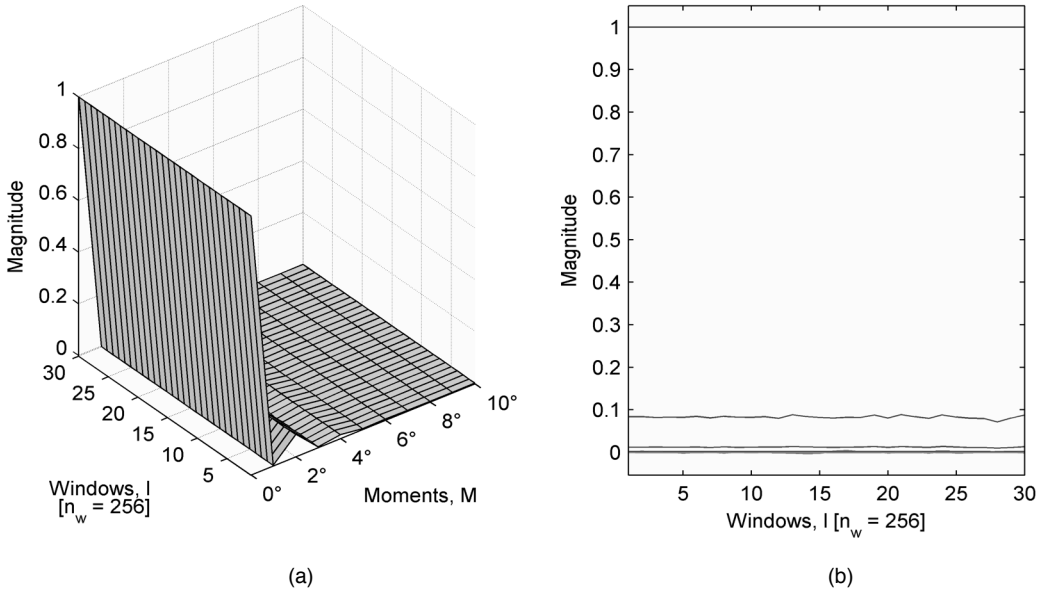


Fig. App.A.11. **(a)** 3D representation of the first ten moments analysis. **(b)** 2D representation of the first ten moments analysis. The window length is $n_w = 2^8$ for a time series with uniform distribution in both cases.

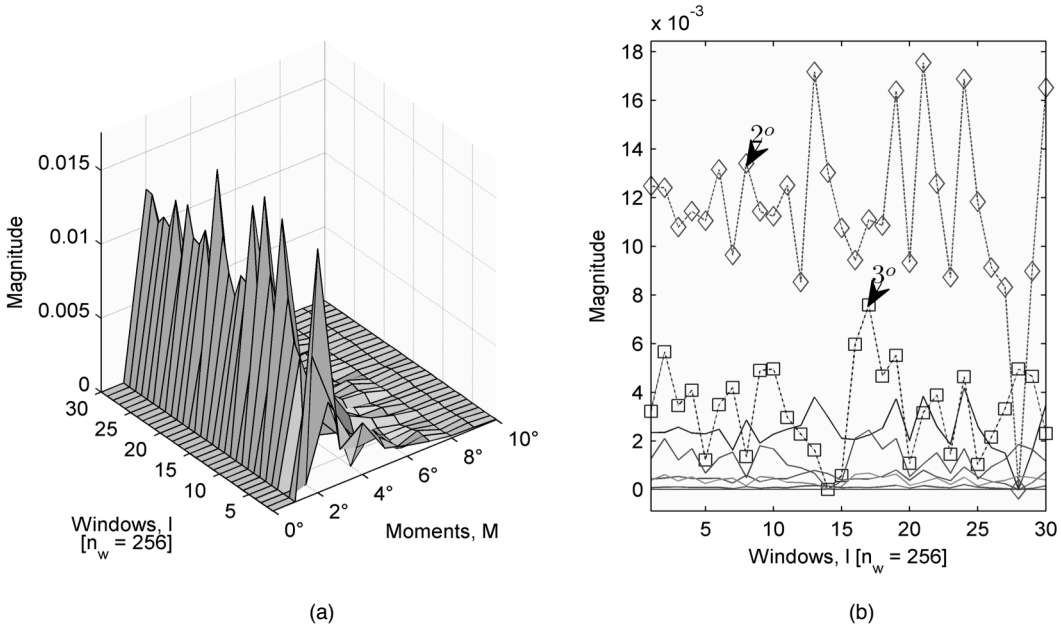


Fig. App.A.12. **(a)** 3D difference representation of the first ten moments analysis. **(b)** 2D difference representation of the first ten moments analysis. The window length is $n_w = 2^8$ for a time series with uniform distribution in both cases.

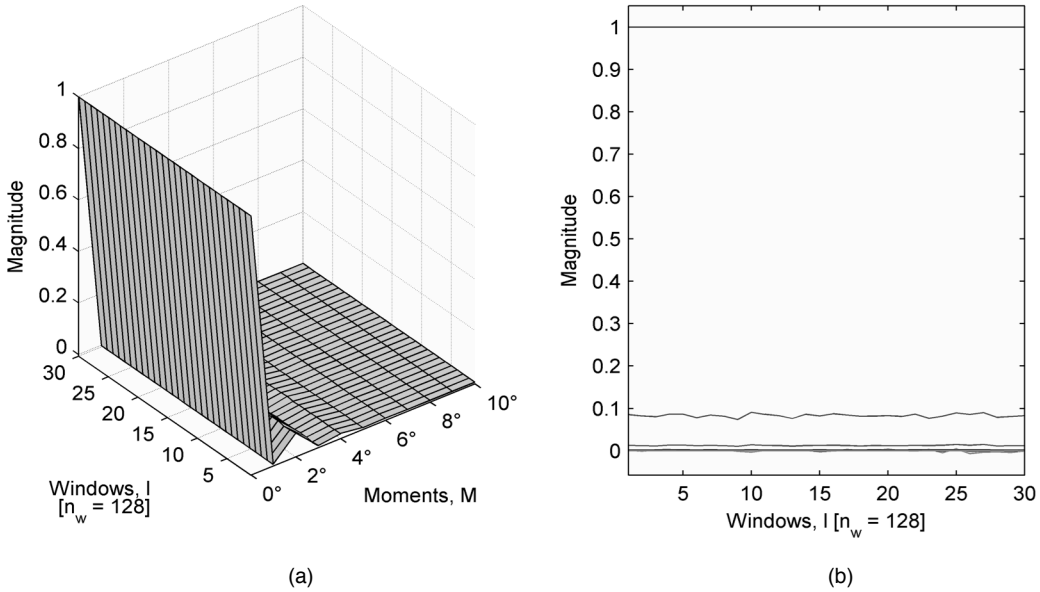


Fig. App.A.13. **(a)** 3D representation of the first ten moments analysis. **(b)** 2D representation of the first ten moments analysis. The window length is $n_w = 2^7$ for a time series with uniform distribution in both cases.

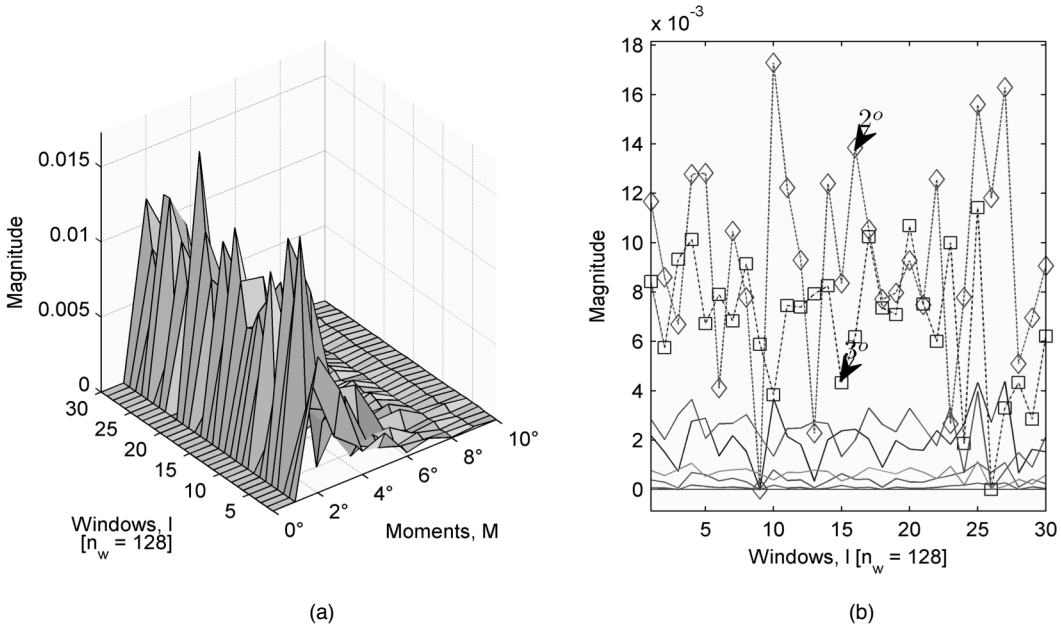


Fig. App.A.14. **(a)** 3D difference representation of the first ten moments analysis. **(b)** 2D difference representation of the first ten moments analysis. The window length is $n_w = 2^7$ for a time series with uniform distribution in both cases.

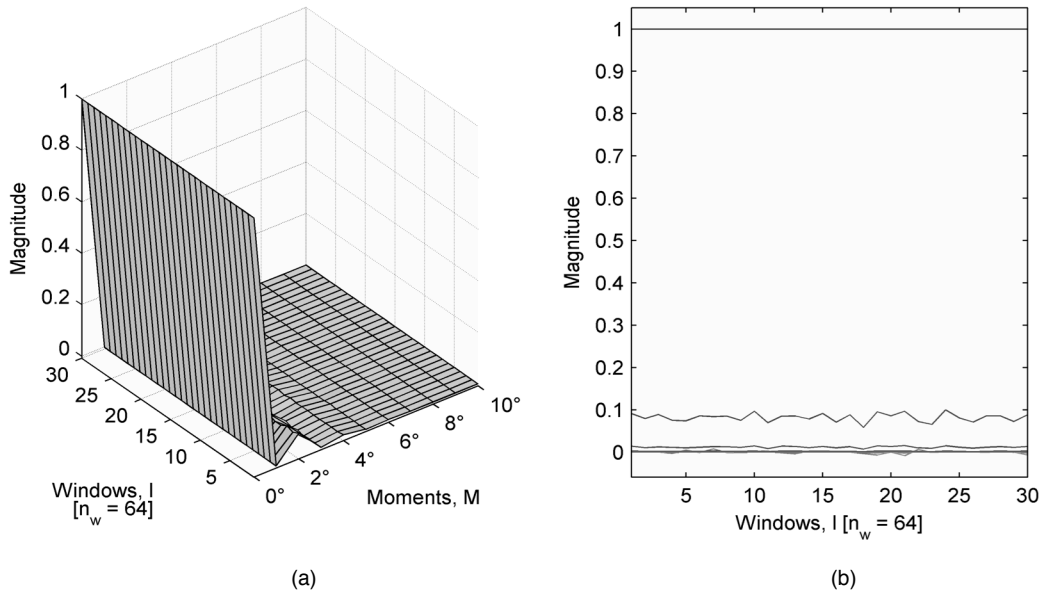


Fig. App.A.15. **(a)** 3D representation of the first ten moments analysis. **(b)** 2D representation of the first ten moments analysis. The window length is $n_w = 2^6$ for a time series with uniform distribution in both cases.

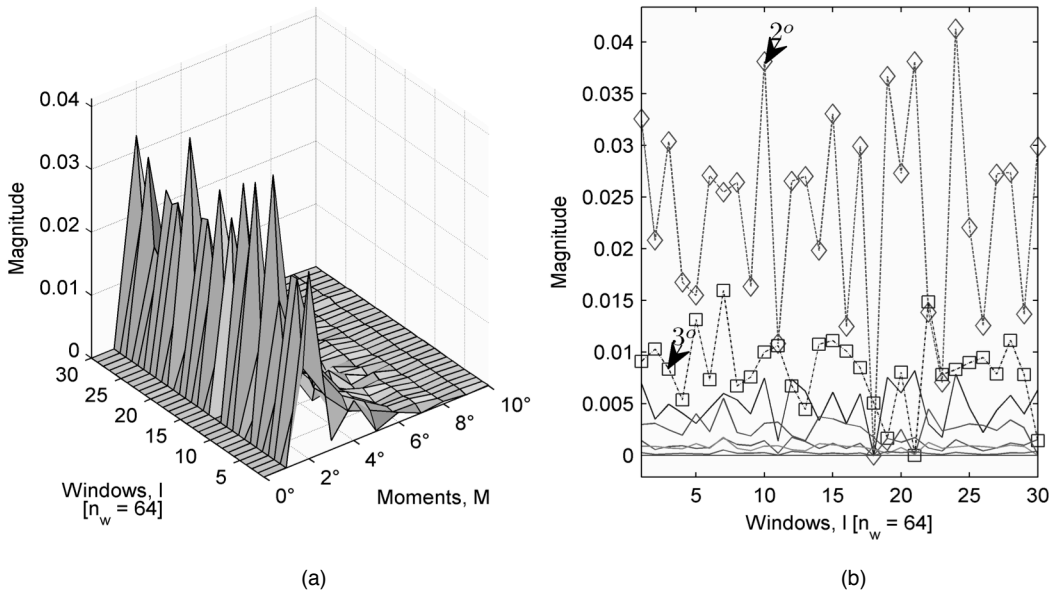


Fig. App.A.16. **(a)** 3D difference representation of the first ten moments analysis. **(b)** 2D difference representation of the first ten moments analysis. The window length is $n_w = 2^6$ for a time series with uniform distribution in both cases.

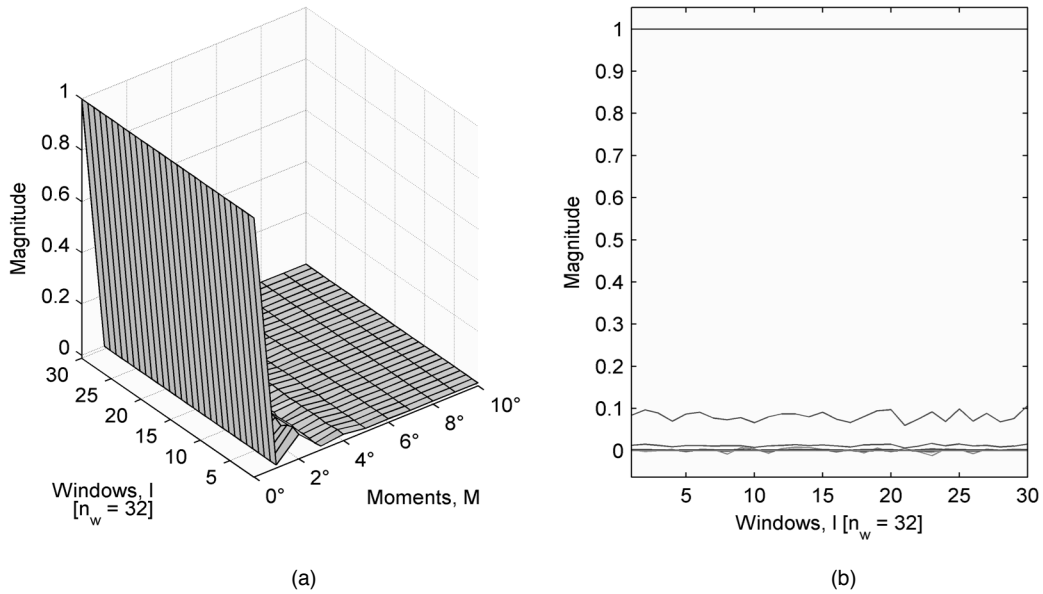


Fig. App.A.17. **(a)** 3D representation of the first ten moments analysis. **(b)** 2D representation of the first ten moments analysis. The window length is $n_w = 2^5$ for a time series with uniform distribution in both cases.

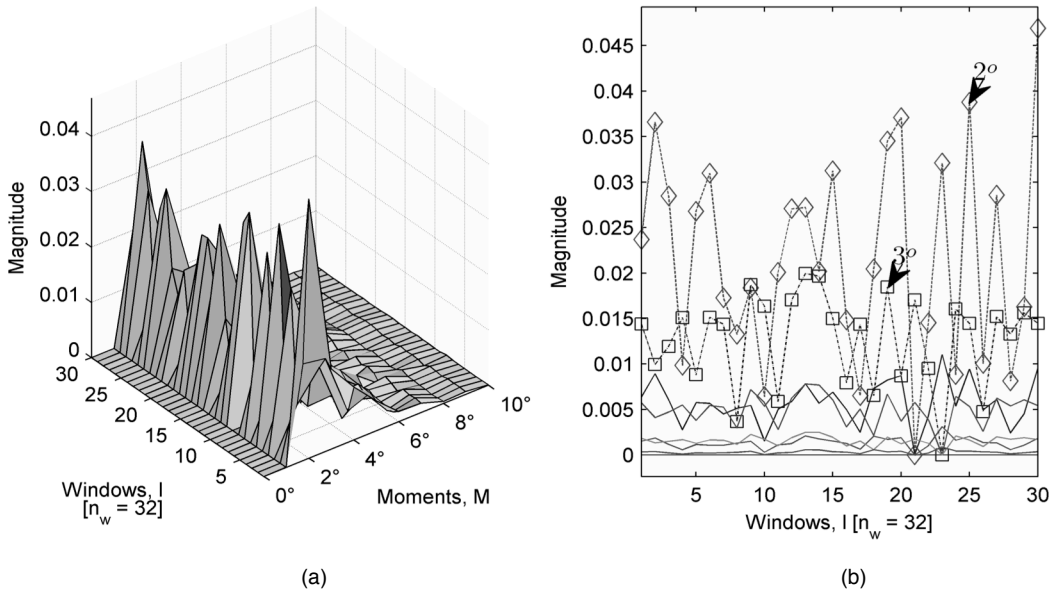


Fig. App.A.18. **(a)** 3D difference representation of the first ten moments analysis. **(b)** 2D difference representation of the first ten moments analysis. The window length is $n_w = 2^5$ for a time series with uniform distribution in both cases.

APPENDIX B

FSS10 TESTING

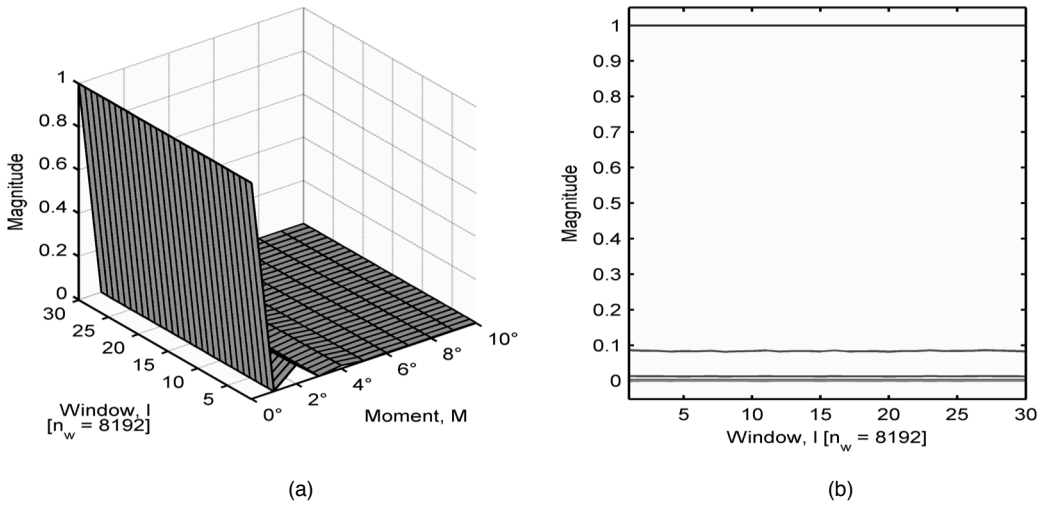


Fig. App.B.1. **(a)** 3D representation of the first ten moments analysis. **(b)** 2D representation of the first ten moments analysis. The window length is $n_w = 2^{13}$ for a time series produced by the cryptosystem in both cases.

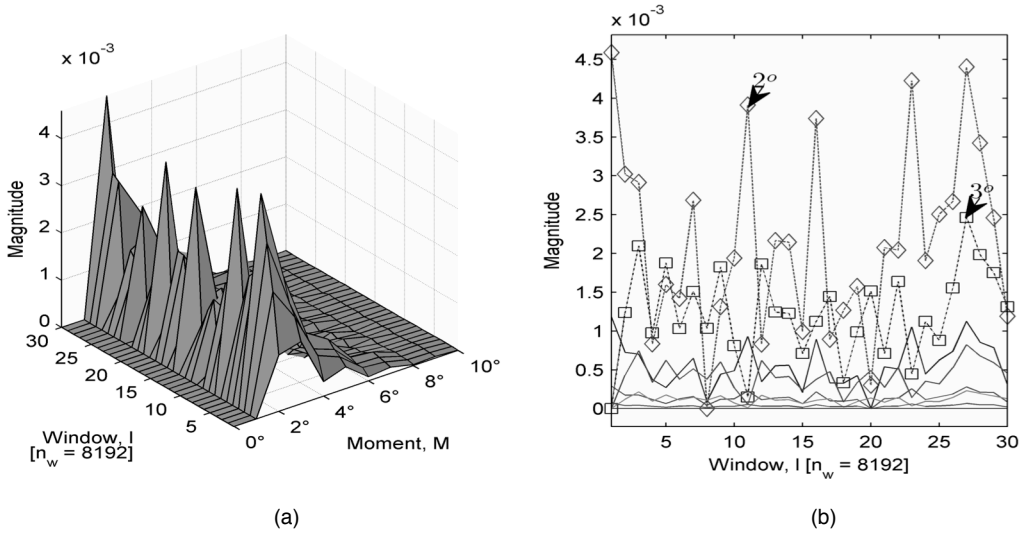


Fig. App.B.2. **(a)** 3D difference representation of the first ten moments analysis. **(b)** 2D difference representation of the first ten moments analysis. The window length is $n_w = 2^{13}$ for a time series produced by the cryptosystem in both cases.

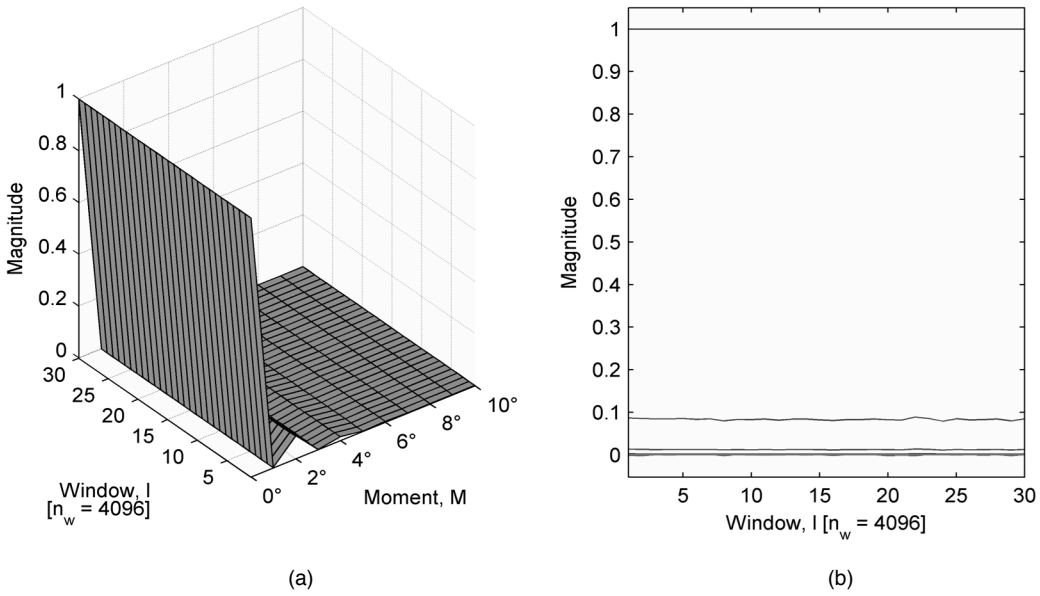


Fig. App.B.3. **(a)** 3D representation of the first ten moments analysis. **(b)** 2D representation of the first ten moments analysis. The window length is $n_w = 2^{12}$ for a time series produced by the cryptosystem in both cases.

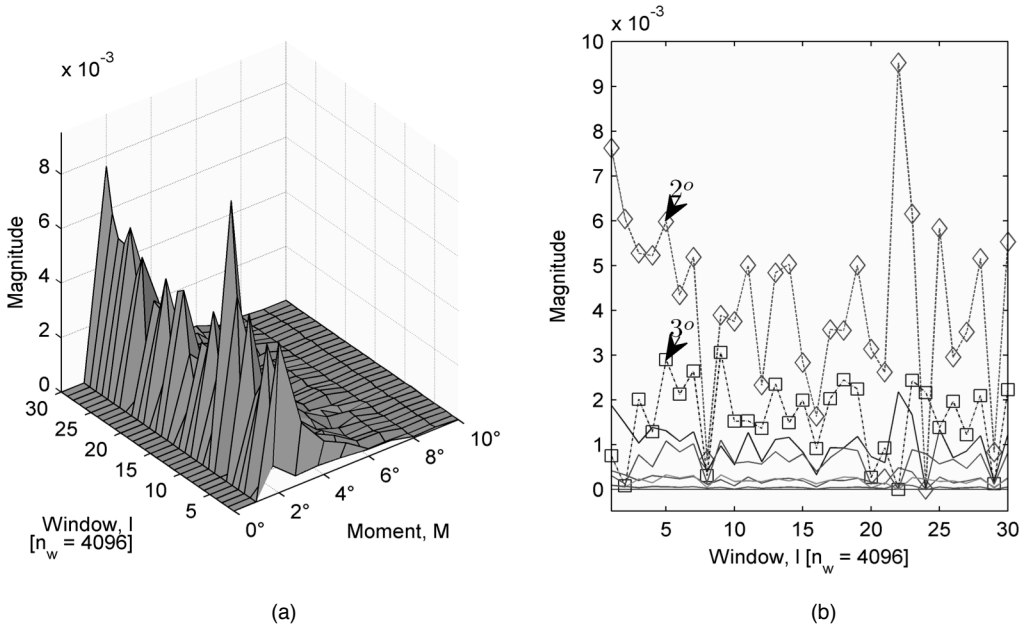


Fig. App.B.4. **(a)** 3D difference representation of the first ten moments analysis. **(b)** 2D difference representation of the first ten moments analysis. The window length is $n_w = 2^{12}$ for a time series produced by the cryptosystem in both cases.

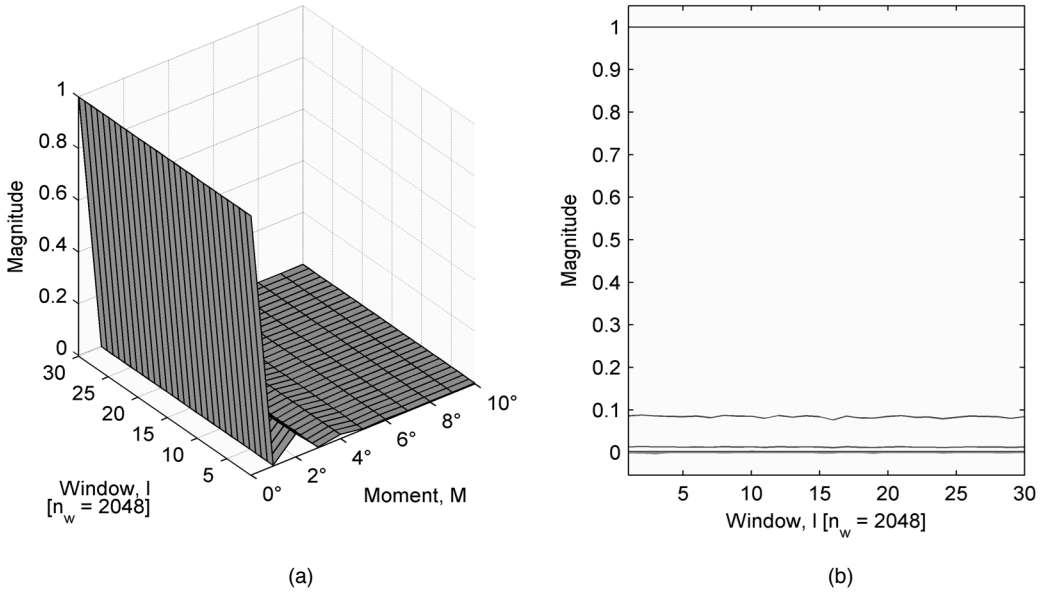


Fig. App.B.5. **(a)** 3D representation of the first ten moments analysis. **(b)** 2D representation of the first ten moments analysis. The window length is $n_w = 2^{11}$ for a time series produced by the cryptosystem in both cases.

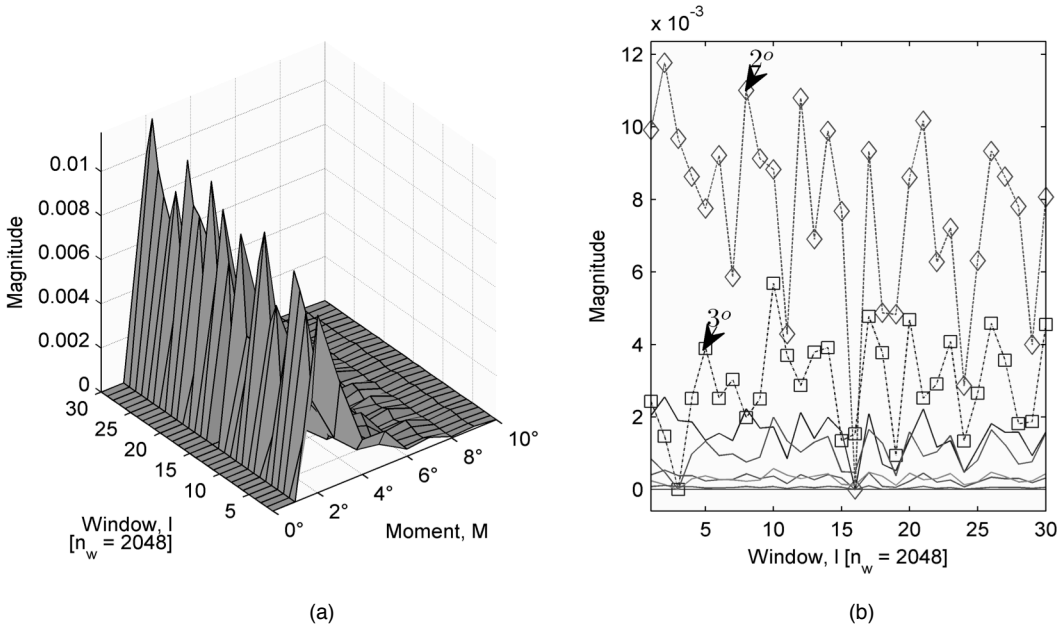


Fig. App.B.6. **(a)** 3D difference representation of the first ten moments analysis. **(b)** 2D difference representation of the first ten moments analysis. The window length is $n_w = 2^{11}$ for a time series produced by the cryptosystem in both cases.

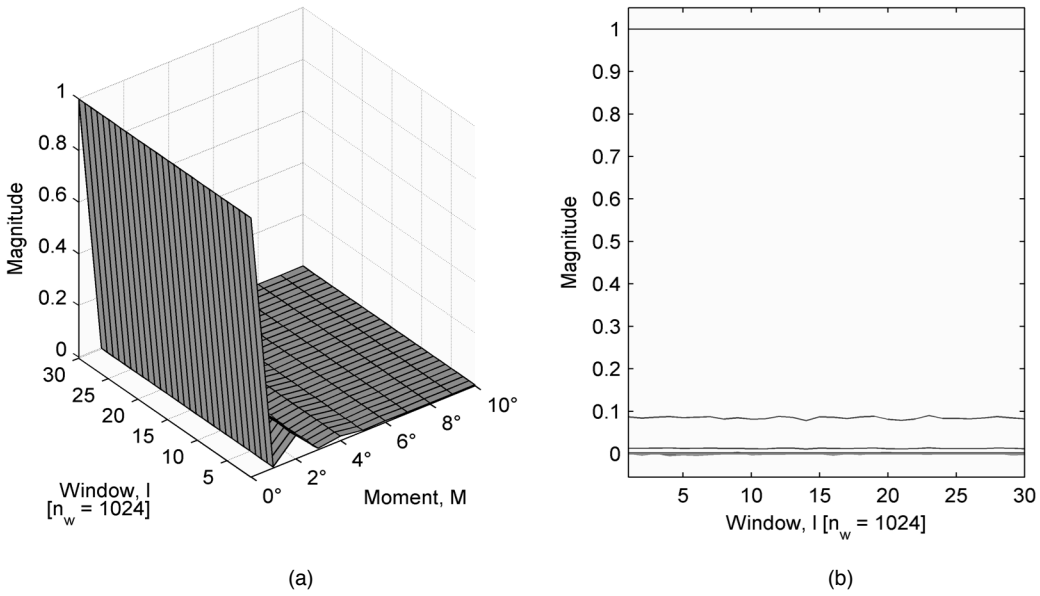


Fig. App.B.7. **(a)** 3D representation of the first ten moments analysis. **(b)** 2D representation of the first ten moments analysis. The window length is $n_w = 2^{10}$ for a time series produced by the cryptosystem in both cases.

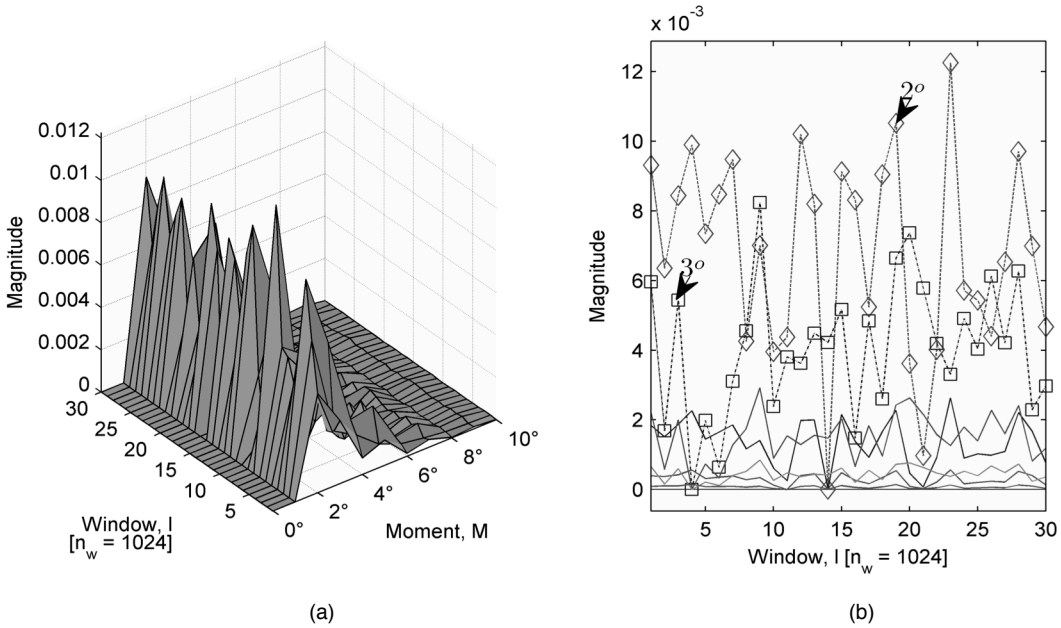


Fig. App.B.8. **(a)** 3D difference representation of the first ten moments analysis. **(b)** 2D difference representation of the first ten moments analysis. The window length is $n_w = 2^{10}$ for a time series produced by the cryptosystem in both cases.

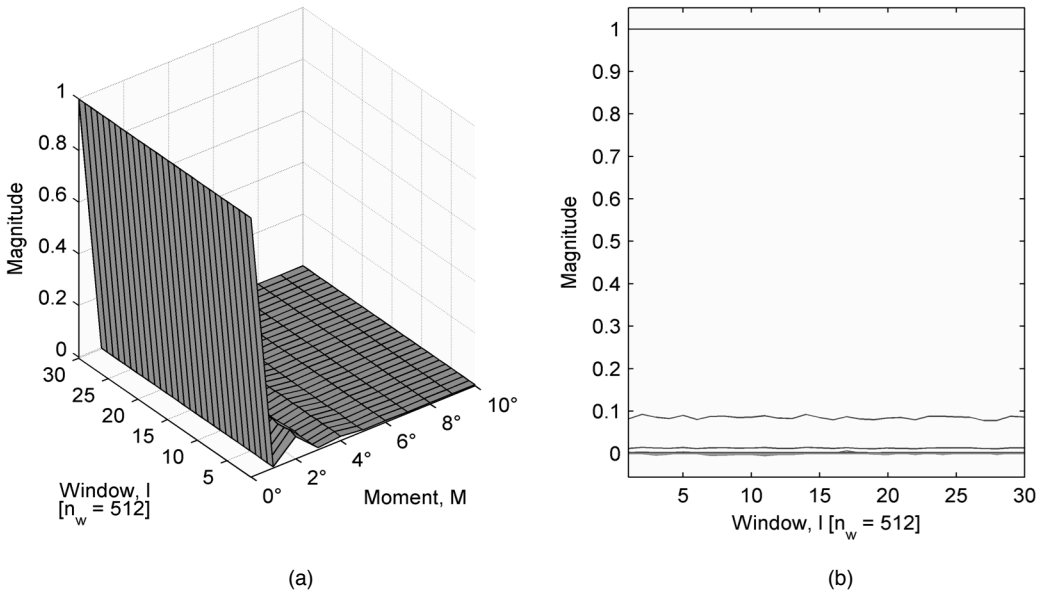


Fig. App.B.9. **(a)** 3D representation of the first ten moments analysis. **(b)** 2D representation of the first ten moments analysis. The window length is $n_w = 2^9$ for a time series produced by the cryptosystem in both cases.

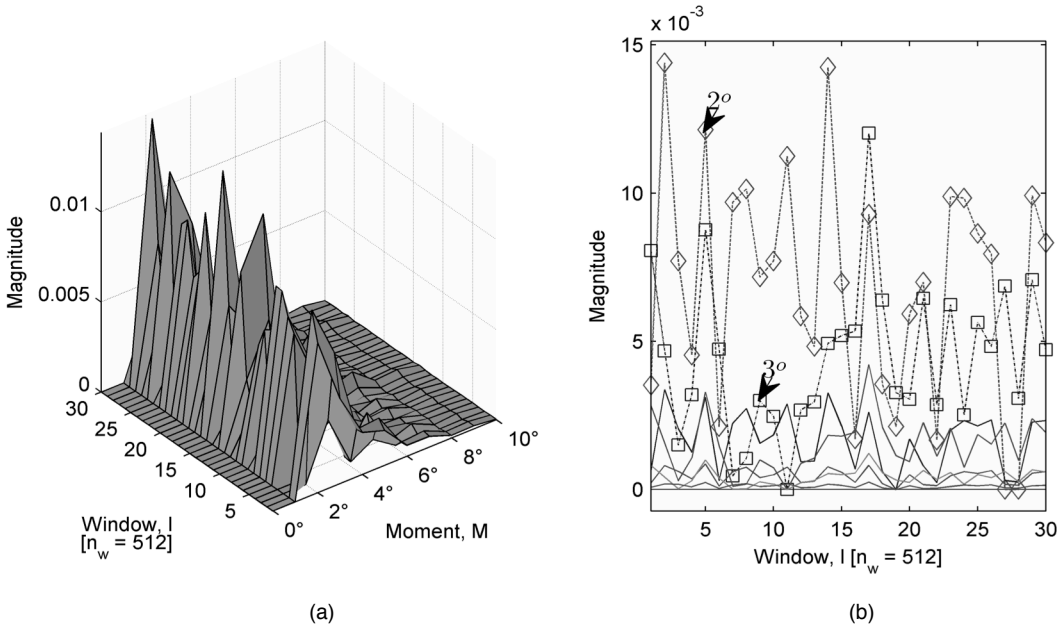


Fig. App.B.10. **(a)** 3D difference representation of the first ten moments analysis. **(b)** 2D difference representation of the first ten moments analysis. The window length is $n_w = 2^9$ for a time series produced by the cryptosystem in both cases.

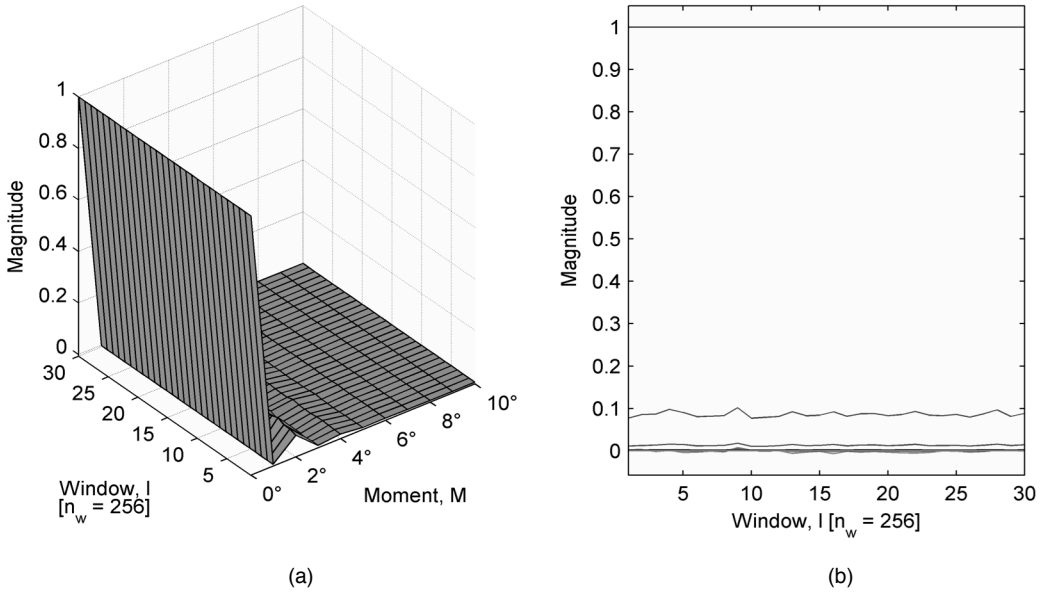


Fig. App.B.11. **(a)** 3D representation of the first ten moments analysis. **(b)** 2D representation of the first ten moments analysis. The window length is $n_w = 2^8$ for a time series produced by the cryptosystem in both cases.

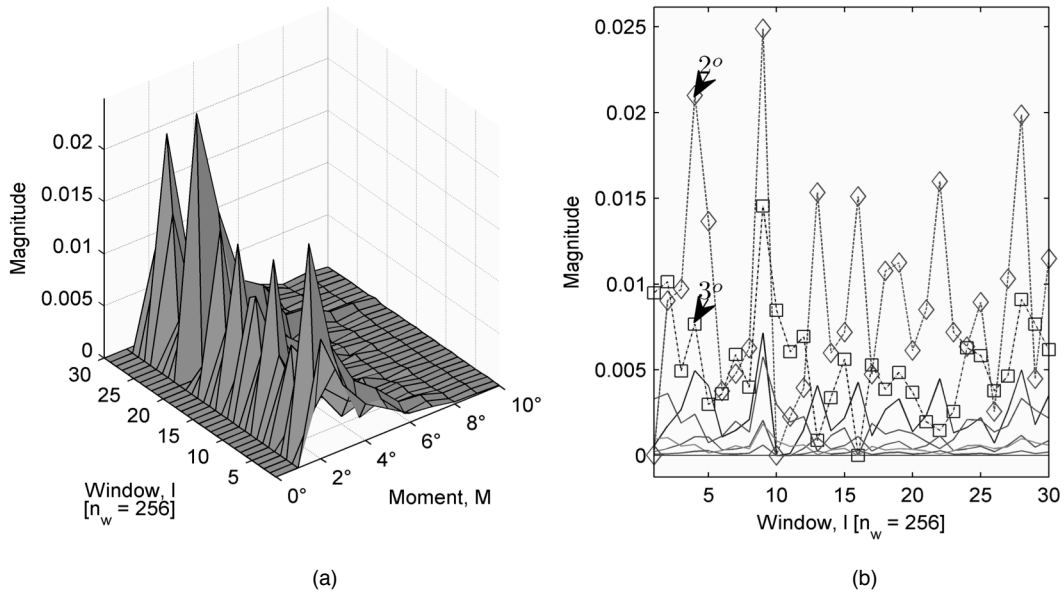


Fig. App.B.12. **(a)** 3D difference representation of the first ten moments analysis. **(b)** 2D difference representation of the first ten moments analysis. The window length is $n_w = 2^8$ for a time series produced by the cryptosystem in both cases.

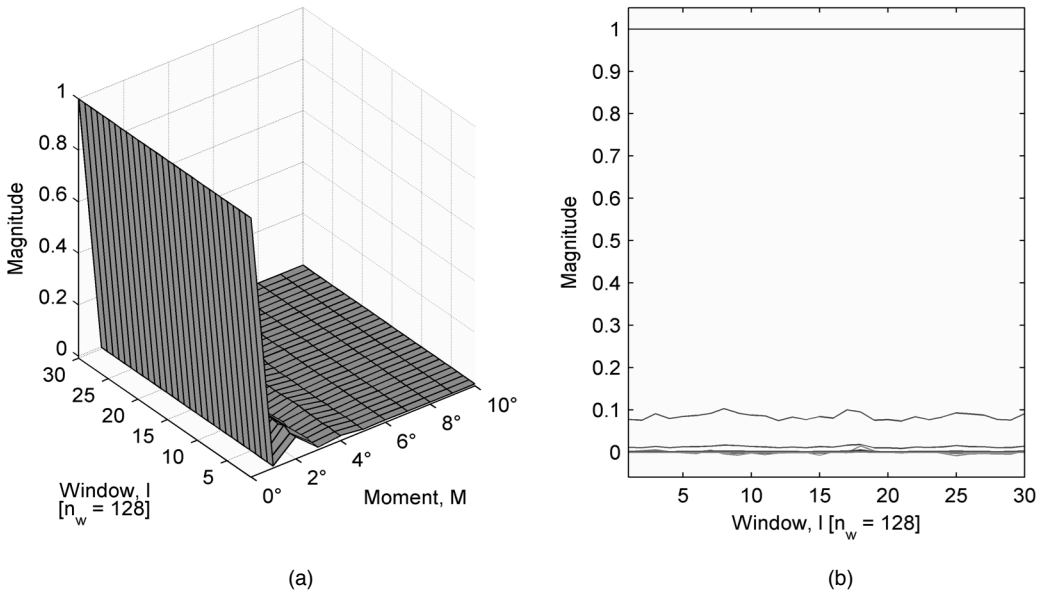


Fig. App.B.13. **(a)** 3D representation of the first ten moments analysis. **(b)** 2D representation of the first ten moments analysis. The window length is $n_w = 2^7$ for a time series produced by the cryptosystem in both cases.

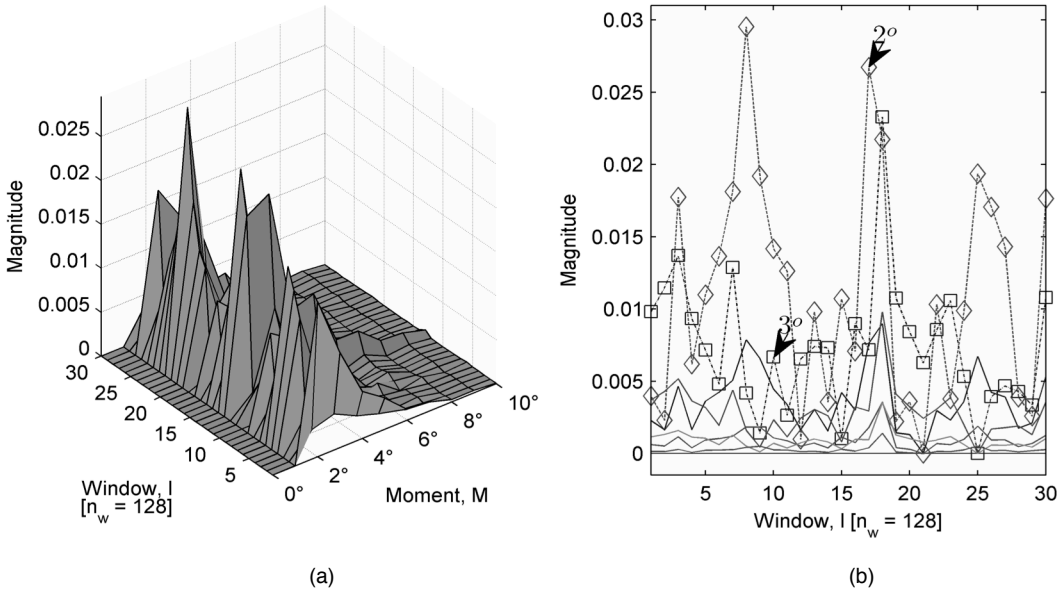


Fig. App.B.14. **(a)** 3D difference representation of the first ten moments analysis. **(b)** 2D difference representation of the first ten moments analysis. The window length is $n_w = 2^7$ for a time series produced by the cryptosystem in both cases.

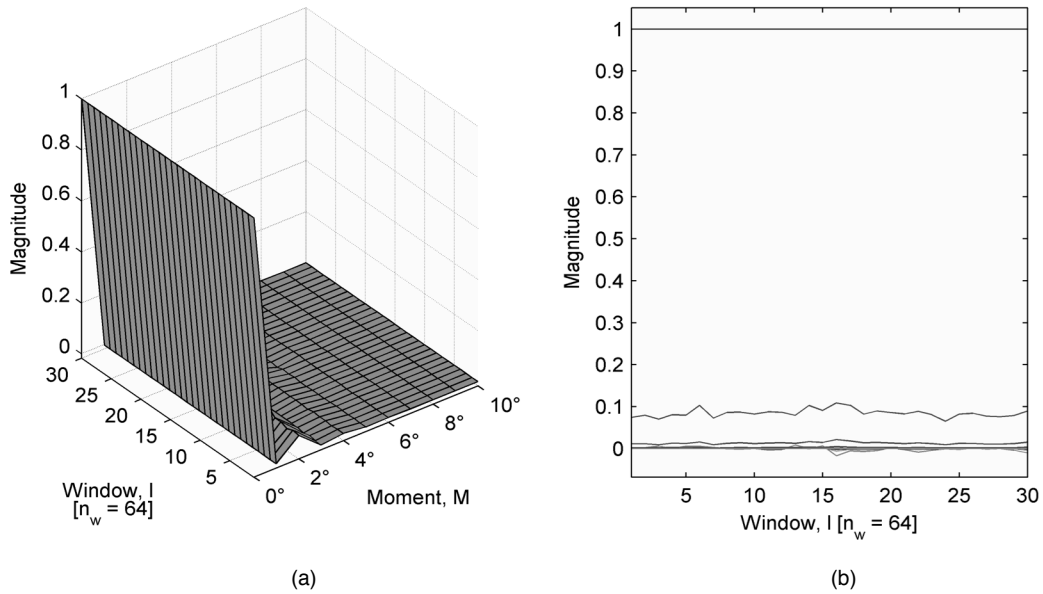


Fig. App.B.15. **(a)** 3D representation of the first ten moments analysis. **(b)** 2D representation of the first ten moments analysis. The window length is $n_w = 2^6$ for a time series produced by the cryptosystem in both cases.

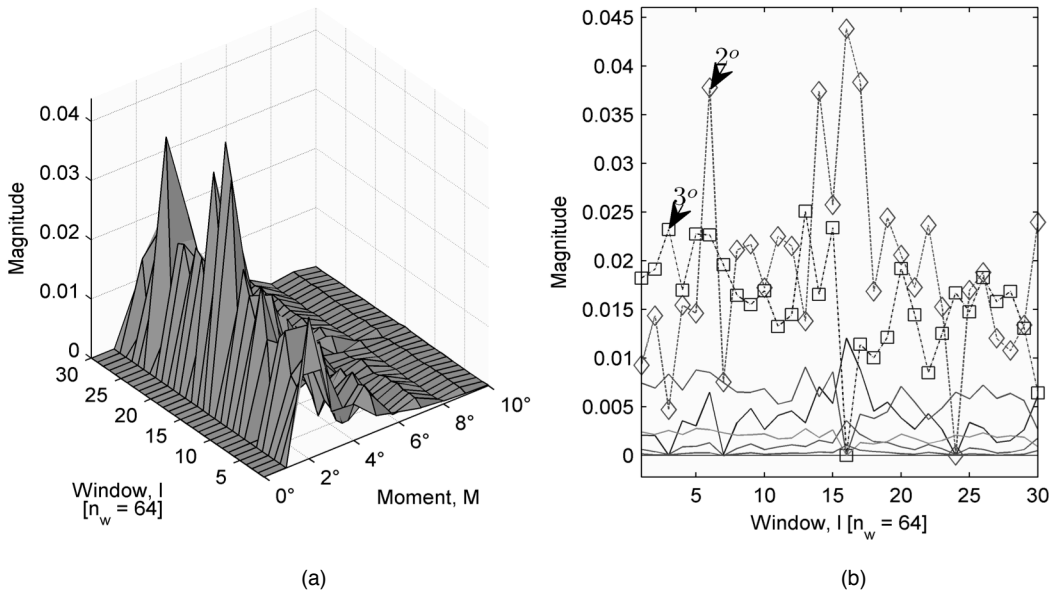


Fig. App.B.16. **(a)** 3D difference representation of the first ten moments analysis. **(b)** 2D difference representation of the first ten moments analysis. The window length is $n_w = 2^6$ for a time series produced by the cryptosystem in both cases.

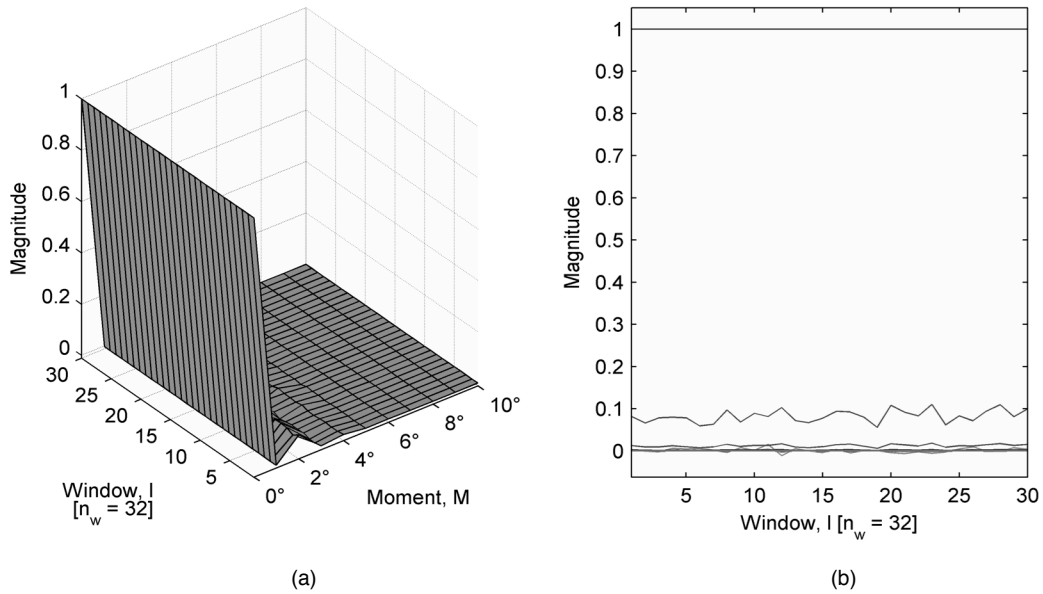


Fig. App.B.17. **(a)** 3D representation of the first ten moments analysis. **(b)** 2D representation of the first ten moments analysis. The window length is $n_w = 2^5$ for a time series produced by the cryptosystem in both cases.

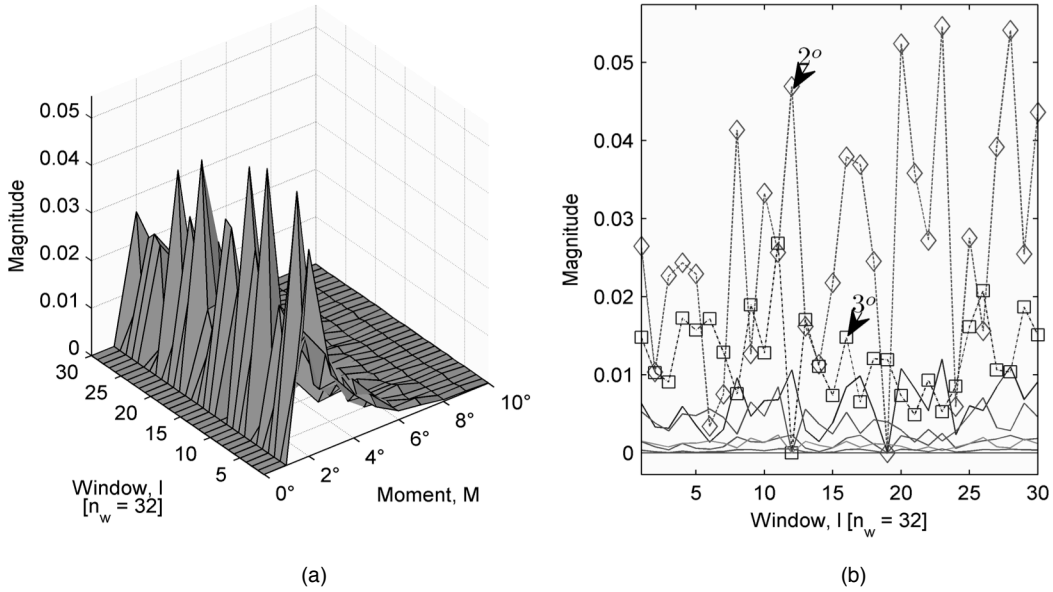


Fig. App.B.18. **(a)** 3D difference representation of the first ten moments analysis. **(b)** 2D difference representation of the first ten moments analysis. The window length is $n_w = 2^5$ for a time series produced by the cryptosystem in both cases.

APPENDIX C

CRYPTOGRAPHY HISTORY:

EVOLUTION OF THE ART OF THE SECRET WRITING

The art of the secret writing or secure communications has been used throughout history since ancient era. The oldest known deliberate transformation of writing dates goes back 4,000 years and it is found in the tomb of the nobleman Khnumhotep II at Menet Khufu, Egypt. This was during the reign of the pharaoh Amenemhet. In ancient India, basic forms of secret communications were also known and practiced. India influenced the development of the cryptography in Thailand where a system known as *the hermit methamorphosing letters* that writes the text backwards is found. The Hebrews applied a form of protocryptography to the Holy Scriptures, but the element of secrecy is lacking.

The Greek incursions are vast. Homer, the greatest epic poet, makes reference to a secret writing in the form of concealment in *The Iliad*. Herodotus, the father of History, includes early steganography methods in *The Histories*. He narrates how the Greeks got information about the Persian king Xerxes plans to conquer them by Demaratus [Sing99]. It was the Spartans who established the first system of military cryptography around the fifth century B.C. This employed the *skytale*. The signalling system based on conversions of letters to numbers devised by Polybious.

The Roman emperor Julius Caesar used an alphabet cipher based on substitutions in the *Gallic Wars*. It was used in politics and military applications. It is noticeable that modern

cryptology springs from the Latin alphabet. After the collapse of the Roman Empire the systems used were simple in the extreme. The most advanced system substituted special signs for letters. Some simple letter substitution based manuscripts appear in Russia in the twelfth and thirteenth century. Then, cryptology stagnated for almost a thousand years, from before 500 to 1400.

Cryptanalysis, the science of unscrambling a message without knowledge of the key, was known by the Arabs. They wrote some cryptanalytic methods after their civilization reached a sufficiently sophisticated level in mathematics, statistics, and linguistics. One of them was capable of braking monoalphabetic substitution ciphers [Sing99]. Arab mathematics delivered the word *cipher* to the world. Cryptanalysis rests in two phenomena: (i) all letters are not used equally in any language and (ii) the proportions in which the letters occur remain constant. The earliest known description of the technique is by the ninth century scientist Abū Yūsūf Ya'qūb ibn as-Sabbāh ibn 'omrān ibn Ismaīl al Kindī better known as "the philosopher of the Arabs." The Islamic administration system relied on secure communication achieved through encryption. Some Arab administrative manuals, such as the tenth-century *Adab al-Kuttāb* ("The Secretaries' Manual"), include sections devoted to cryptography [Sing99]. Cryptography Arabic knowledge was fully set in the cryptology section of the encyclopedia *Subh al-a 'sha* completed in 1412. This section was composed of two parts: one dealing with symbolic actions and allusions, the other with invisible inks. The first known cipher that provides more than one substitute for a PT letter is encountered here.

The first known European book that describes the use of cryptography was written in the thirteen century by the English Franciscan monk and polymath Roger Bacon. *Epistle on the Secret Works of Art and the Nullity of Magic* contained seven methods for messages secrecy

[Sing99]. A work by Geoffrey Chaucer from the fourteenth century, *Treatise on the Astrolabe*, shows early European encryption. After the revival in the arts, sciences, and scholarship in the beginning of the fifteenth century during the Renaissance nurtured the capacity for cryptography and provided solid ground for diplomacy [Sing99]. From about 1400 to about 1850, a system known as *nomenclator* (an example of this is a cipher used by Luis XIV of France) that combined a codebook with homophonic substitution tables dominated cryptography. The Italian Leon Battista Alberti invented a *polyalphabetic* cipher during the fifteenth century. This was a critical advance in cryptology but it was not used by the major world powers until the end of the nineteenth century. Most of today's cryptosystems are based in this idea. The German Johannes Trithemius revealed the square table also known as *tabula recta* in 1518. This is a system that reveals all possible CTs in a given alphabet. The Italian Giovan Battista Bellaso proposed a system based in two critical steps, (i) a literal easily remembered and (ii) an easily changed key called *countersign*, in his book *La cifra* in 1553. With this system, different ambassadors could use individual keys. This provides an archetype of the public key cryptography scheme used widely today. The French Blaise de Vigenère invented the *autokey* system in 1586. This uses the plain text as the key. After its reinvention, it was used in cryptology again late in the nineteenth century. Vigenère's work culminated in his *Traictè des Chiffres* ("A Treatise on Secret Writing"), published in 1586. The nomenclator was preferred over the polyalphabetic ciphers due to their slowness and especially in the autokey based if a mistake was done in the encipherment process the message is lost.

No sign of sustained cryptanalysis techniques are found in the secret writing history until the sixteenth century. Just occasional cases were spotted. The dark side of cryptology which is

cryptanalysis did not exist but cryptography was growing. The sphere of cryptology just contained a half, cryptography, with its complement about to rise. Giovanni Soro from the Venice organization *The Council of Ten* was one of the greatest cryptanalysts around the first half of the sixteenth century. Even the papal curia sent him ciphers that no one in Rome could solve. Henry IV king of France at the end of the sixteenth century and Mary I of Scotland had lives influenced by the cryptanalysis work of Francois Viete and Thomas Phelippes respectively. The first case intended to save Henry IV crown and the second lead to Mary I death. Not for the first time, a life hung on the strength of a cipher [Sing99]. The French greatest cryptologist, Antointe Rossignol, solved cryptograms for Henry II of Bourbon when fighting the Huguenots in 1628. This called the attention of the Cardinal Richeliu and he recruited him. After this, Rossignol helped to block the expected help by the Huguenots from an English fleet and started to serve in the royal service. A cipher designed by Antoine and Bonaventure Rossignol which after two centuries still locked French secrets. It was until 1890 when Victor Grendon and the Commandant Ètienne Bazeries attempted to decipher them. Bazeries found traps that the Rossignols laid within the cipher (*e.g.*, one number represented neither a syllable nor a letter, but instead deviously deleted the previous number) [Sing99].

The telegraph by Samuel F. B. Morse in 1844, a Sputnik like invention started a new revolution in cryptography and started to furnish it. Commercial codes like Smith's afforded sufficient security for most business precluding message sight comprehension. Nomenclators used in that time by Government ministries were composed from two to fifty thousand codes especially in high-level military and diplomatic cryptography. Secrecy became extremely important for commanders exerting instantaneous and continuous control over masses of men

deployed over large areas in tactical operations. Here the Vigenère system enters into the military communication systems. The nomenclator reign of 450 years was broken. Nevertheless, Charles Babbage succeeded in breaking the Vigenère system around 1854. By this, he made the greatest breakthrough in cryptanalysis since the Arab scholars of the ninth century broke the monoalphabetic cipher by means of the frequency analysis. In Babbage case this was done by sheer cunning to show Thwaites that the cipher he was trying to patent was not new [Sin99]. Cryptanalysis methods arranged ciphers according to empirical complexity measures. The *wheel cipher* invented by Thomas Jefferson before the telegraph is an interesting polyaphabetical example that confers him the title of Father of American Cryptography. This cipher was rediscovered in the Library of Congress in the beginning of the nineteenth century and started to be used. The first digraphic cipher was invented by Wheatstone and popularized by Playfair in 1854. This undercuts the monographic frequency cryptanalytic methods used until then. Pliny Earle Chase proposed fractionating cipher systems in 1859. The problem of achieving a general solution for polyalphabetic ciphers with repeated keywords that displeased cryptanalysts by more than 300 years was answered in *Die Geheimschriften und die Dechif-frir-kunst* published by the Polish Friederich W. Kasiski in 1863. The *one-time pad* which has been proven impossible to crack if used correctly was described by Frank Miller in 1882. *La Cryptographic militaire* published by the French Auguste Kerckhoffs in 1883 is one of the most concise books written. He was the first person in identify military cryptosystems requirements based on simplicity, reliability and rapidity: (i) theoretically unbreakable or at least in practice; (ii) its compromise should not compromise its users; (iii) rememberable and easily changeable key; (iv) transmissible cryptograms by telegraph; (v) portability; and (vi) operation by a single person

requiring no deep knowledge about it. Here it is also stated the Kerckhoffs' principle: "The security of a cryptosystem must not depend on keeping secret the cryptoalgorithm. The security depends only on keeping secret the key [Sing99]." During the American Revolution in the latest half of the eighteenth century, George Washington used a codebook to collect intelligencse about the British forces and their movements.

One of the biggest human creations in 1895, the radio, magnified the chief military advantage of telegraphy by eliminating wires. It joined through the ether armies, naval, and air forces. Easy eavesdropping also emerged given the omnidirectional nature of radio transmissions. The auxiliary and academic characteristics of cryptanalysis during the telegraph golden age were transformed into a weapon character by the radio. The possibility of constant interception was born. Radio created modern cryptanalysis and let it stand at the same level of cryptography revolutionized by the telegraph.

World War I provided maturity to cryptanalysis. The cryptanalytic team operating in Room 40 in the Almiralty (authority responsible for the Royal Navy command) in England, which was headed by the First Lord Winston Churchill, intercepted and solved 15,000 German secret communications from 1914 to 1919. This group maintained England ahead of its enemies through the war. Room 40's solution of an enemy message propelled the United States into WWI. This enabled the Allies to win. This exposes cryptanalysis consequences in history. Other cryptanalyst groups created during WWI are G.2 A.6 and MI-8.

Most of the basic scrambler systems were invented during the 1920s and 1930s by engineers working for the growing radio and telephone companies (*e.g.*, American Telephone and Telegraph Company).

During World War II different electromechanical cryptographic oriented devices appeared: the red and purple Japanese ciphers, the German enigma, the SIGABA from America, among others. These machines provided the codenames for important operations and protected their secret projects giving a big boost to cryptography, given their huge power to change and scramble text [MePS11]. In 1932, Thomas H. Dyer, became the father of machine cryptanalysis when he installed IBM machines to speed up solution. He led the cryptanalyst group in Station HYPO in Hawaii from 1936 to 1945. This group was responsible for most of the breakthroughs in reading Japanese naval communications during the war in the Pacific. Section C in the Signal Intelligence Service, organism created by William Frederick Friedman, devised hundreds of ciphers, thousands of keys lists. It printed 5,000,000 classified documents. It tested the security of Army chipper machines (*e.g.*, SIGABA) by attempting to solve them. WWII enlarged, accelerated, and intensified the cryptology body provided by the changes introduced by telegraphy and radio. This is reflected in the U.S. Army and Navy increase of 400 persons in cryptology in WWI to 16,000 in WWII. It was in the actions of WWII when cryptology became the most important source of secret intelligence for a nation. The Russians cracked the Enigma in 1942. At this time the Soviet Union guarded diplomatic flanks using the one-time pad. By this her crucial messages where neither read by foes, neutral, nor allies. This cryptosystem is used until today fearing nothing from cryptanalysis.

A form of stenography that became popular during WWII was the microdot. This was a technique used by German agents in Latin America. It consisted of photographically shrink a page of text down to a dot less than one millimiter then hiding it on top of a full stop. The FBI spotted the first microdot in 1941 [Sing99].

During the cold war Russia solved ciphers in use at the American embassy in Moscow demonstrating a profound understanding of cryptography and cryptanalysis. With the mind blowing amounts of traffic for cryptanalysis as a base, the greatest cryptologic organization in history which is the National Security Agency (NSA), was created in 1952. Technological developments from the transistor to highly specialized computing schemes, tools as Galois field theory, stochastic processes, matrix, and number theory are used inside its walls. The weakest radio messages one can think off can be analysed by the NSA.

The greatest treatise authored by al Kindī, the philosopher of the Arabs, was rediscovered in 1987 in the Sulaimaniyyah Ottoman Archive in Istanbul. It is entitled A Manuscript on Deciphering Cryptographic Messages. It encapsulates cryptanalysis techniques from the ninth century in the two short paragraphs [Sing99]:

One way to solve an encrypted message, if we know its language, is to find a different PT of the same language long enough to fill one sheet or so, and then we count the occurrences of each letter. We call the most frequently occurring letter the “first,” the next most occurring letter the “second,” the following most occurring letter the “third,” and so on, until we account for all the different letters in the PT sample.

Then we look at the CT we want to solve and we also classify its symbols. We find the most occurring symbol and change it to the form of the “first” letter of the PT sample, the next most common symbol is changed to the form of the “second” letter, and the following most common symbol is changed to the form of the “third” letter, and so on, until we account for all symbols of the cryptogram we want to solve.

After this journey of 4000 years in the fascinating history of secret writing the intelligence

collection continue up to this very moment. Today, information is gathered by aircrafts, ships, submarines, satellites, and electronic eavesdropping. Here, the need for generating and improving data protection tools.

APPENDIX D

QUANTUM CRYPTOGRAPHY

The basis of classical computing is the common sense notion that a low potential state and a high potential state are mutually exclusive, so both of them cannot occur simultaneously. This way, two consecutive and independent operations on those bits will happen normally, through two consecutive logic steps. This does not apply in quantum mechanics [MePS11].

In 1984, quantum cryptography (QC) was introduced by Bennett and Brassard when they proposed the protocol for secret key distribution known as BB84 [BrBe84]. This protocol is the most commonly analysed and implemented. Quantum cryptography ensures the confidentiality of information transmitted between two parties exploiting elementary particles behaviour (*e.g.*, photons). The most mature application of quantum information science is the distribution of secret keys and surprisingly it does not require a quantum computer. It uses only a quantum (*e.g.*, fiber optics and/or lasers) and a classic (*e.g.*, e-mail, telephone, or radio waves) communication channel. Quantum key distribution protocols have the final goal of creating a common key for cryptographic applications in symmetric schemes [MePS11].

The Heisenberg uncertainty principle is the basis for QC. It is not possible to know the location and speed of a particle at the same time. The information storage medium is at the atomic scale in the form of q-bits (associated to a probability distribution which indicates the chances of finding each possible value when measuring) analogous to traditional bits. A q-bit can be either 0 or 1 at the same time by laws of quantum mechanics. This coexistence is called superposition. By the uncertainty principle, an eavesdropper cannot know everything about a

photon carrying a key bit, and part of the information can be destroyed because measurement is destructive in quantum mechanics. Detection of eavesdroppers is possible because any measurement performed on the particle carrying information disturbs it. The value of an eavesdropper capturing a single q-bit and not altering it is known to have a probability of ¾. This value comes from the sum of ½ that implies using one of the two available basis (if the basis used by the eavesdropper the q-bit is not altered otherwise it will be) and ¼ from the receiver returning the correct q-bit value (the receiver has a probability of ½ of ½ to return a correct value). The probability of not changing q-bits values diminishes when the number of q-bits under analysis increases.

Quantum key distribution is achievable using current technologies (*e.g.*, lasers and fiber optics, and electronic noise sources). Single photon sources (*e.g.*, trapped atoms or ions as nitrogen vacancy colour center in diamonds) are becoming within reach of current technologies. To accomplish it in QC, a source of truly random numbers (*i.e.*, noise in a resistor) and the classical authenticated channel are needed in the creation of the secret key.

Single photon polarization in a binary way is possible in quantum mechanics. Therefore light polarization can be understood as a quantum property represented as a vector in a bi-dimensional space [MePS11].

Quantum states and quantum pieces of information are written using vertical-horizontal $(\mathrm{V}-\mathrm{H})$ and diagonal-counter diagonal $(\mathrm{D}-\mathrm{C})$ orthogonal basis. These basis are represented using Dirac's notation $|0\rangle$ and $|1\rangle$ or $|+\rangle$ and $|-\rangle$ [Assc06]. They also are represented by $\rightarrow$ and $\uparrow$ for the $\mathrm{V}-\mathrm{H}$ case. The basis for $\mathrm{D}-\mathrm{C}$ case uses $\nearrow$ and $\nwarrow$ [BrBe84] and [MePS11]. An emitter has both basis as a base to polarize photons in four different positions. A receiver in the

other hand uses just one of these basis. This provides the following possibilities: (i) if the polarized photon is coded as $|0\rangle$, it is captured exactly as $|0\rangle$; (ii) if the polarized photon is coded as $|1\rangle$, it is captured exactly as $|1\rangle$; (iii) if the emitter and the receiver used different basis the information is lost at the moment that the receiver captures it (*e.g.*, the emitter uses $\mathrm{V}-\mathrm{H}$ and sends a $\uparrow$, then the receiver uses $\mathrm{D}-\mathrm{C}$ to capture the q-bit. The q-bit will be captured as $\nwarrow$ or $\nearrow$), with probability of ½ for capturing a $|0\rangle$ and ½ for capturing a $|1\rangle$ given the fact that the $\mathrm{V}-\mathrm{H}$ and the $\mathrm{D}-\mathrm{C}$ basis are rotated exactly by $\pi/4$. The binary convention in the BB84 protocol is shown in table D2.2.

Table D.1: Binary Convention Used in the BB84 Protocol.

BASIS	***q*-BIT**	
	0	**1**
$\mathrm{V}-\mathrm{H}$	$\uparrow$	$\rightarrow$
$\mathrm{D}-\mathrm{C}$	$\nearrow$	$\nwarrow$