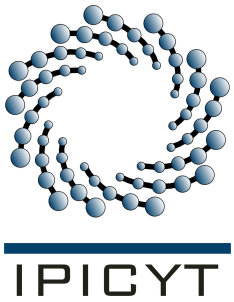


INSTITUTO POTOSINO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA, A.C.

POSGRADO EN CIENCIAS APLICADAS

Estudio de mapeos caóticos discretos y su aplicación en criptografía

Tesis que presenta

Moisés García Martínez

Para obtener el grado de

Doctor en Ciencias Aplicadas

En la opción de

Control y Sistemas Dinámicos

Director de la Tesis:

Dr. Eric Campos Cantón

San Luis Potosí, S.L.P., Enero de 2015

IPICYT

Constancia de aprobación de la tesis

La tesis ***"Estudio de mapeos caóticos discretos y su aplicación en criptografía"*** presentada para obtener el Grado de Doctor en Ciencias Aplicadas en la opción de Control y Sistemas Dinámicos fue elaborada por **Moisés García Martínez** y aprobada el **treinta de enero del dos mil quince** por los suscritos, designados por el Colegio de Profesores de la División de Matemáticas Aplicadas del Instituto Potosino de Investigación Científica y Tecnológica, A.C.

Dr. Eric Campos Cantón
Director de la tesis

Dr. Rubén Vázquez Medina
Jurado en el Examen

Dr. Hugo Cabrera Ibarra
Jurado en el Examen

Dr. Cornelio Posadas Castillo
Jurado en el Examen

Dr. Juan Gonzalo Barajas Ramírez
Jurado en el Examen

Créditos Institucionales

Esta tesis fue elaborada en la División de Matemáticas Aplicadas del Instituto Potosino de Investigación Científica y Tecnológica, A.C., bajo la dirección del Dr. Eric Campos Cantón.

Durante la realización del trabajo el autor recibió una beca académica del Consejo Nacional de Ciencia y Tecnología (No. 217447) y apoyos del Instituto Potosino de Investigación Científica y Tecnológica, A. C.

Este trabajo fue apoyado por el proyecto CONACYT-Fondos Sectoriales-SEP No. 181002.

IPICYT

Instituto Potosino de Investigación Científica y Tecnológica, A.C.

Acta de Examen de Grado

El Secretario Académico del Instituto Potosino de Investigación Científica y Tecnológica, A.C., certifica que en el Acta 019 del Libro Primero de Actas de Exámenes de Grado del Programa de Doctorado en Ciencias Aplicadas en la opción de Control y Sistemas Dinámicos está asentado lo siguiente:

En la ciudad de San Luis Potosí a los 30 días del mes de enero del año 2015, se reunió a las 16:00 horas en las instalaciones del Instituto Potosino de Investigación Científica y Tecnológica, A.C., el Jurado integrado por:

Dr. Rubén Vázquez Medina	**Presidente**	**IPN**
Dr. Eric Campos Cantón	**Secretario**	**IPICYT**
Dr. Cornelio Posadas Castillo	**Sinodal externo**	**UANL**
Dr. Hugo Cabrera Ibarra	**Sinodal**	**IPICYT**
Dr. Juan Gonzalo Barajas Ramírez	**Sinodal**	**IPICYT**

a fin de efectuar el examen, que para obtener el Grado de:

DOCTOR EN CIENCIAS APLICADAS
EN LA OPCIÓN DE CONTROL Y SISTEMAS DINÁMICOS

sustentó el C.

Moisés García Martínez

sobre la Tesis intitulada:

Estudio de mapeos caóticos discretos y su aplicación en criptografía

que se desarrolló bajo la dirección de

Dr. Eric Campos Cantón

El Jurado, después de deliberar, determinó

APROBARLO

Dándose por terminado el acto a las 18:30 horas, procediendo a la firma del Acta los integrantes del Jurado. Dando fe el Secretario Académico del Instituto.

A petición del interesado y para los fines que al mismo convengan, se extiende el presente documento en la ciudad de San Luis Potosí, S.L.P., México, a los 30 días del mes de enero de 2015.

Dr. Marcial Bonilla Marín
Secretario Académico

INSTITUTO POTOSINO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA, A.C.
IPICYT
SECRETARIA ACADEMICA

Mtra. Ivonne Lizette Cuevas Vélez
Jefa del Departamento del Posgrado

Dedicado a mi esposa,
padres y hermanos

Agradecimientos

En primer lugar mi más profundo y sincero agradecimiento a mi director de tesis Dr. Eric Campos Cantón quien desde un principio confió en mí y me brindo la posibilidad de trabajar bajo su tutela, a lo largo de estos años siempre me mostro y sigue mostrando su apoyo incondicional, le estaré eternamente agradecido por esta oportunidad.

De igual forma expreso mi agradecimiento a los revisores de este trabajo de tesis Dr. Alejandro Ricardo Femat Flores, Dr. Rubén Vázquez Medina, Dr. Cornelio Posadas Castillo, Dr. Hugo Cabrera Ibarra y Dr. Juan Gonzalo Barajas Ramírez por sus valiosos comentarios y por todo el tiempo dedicado para la elaboración de este trabajo.

A todos los investigadores que forman parte de la división ya que fueron parte fundamental de mi formación. A la división de Matemáticas Aplicadas y al IPICyT por el apoyo económico brindado para poder asistir a congresos.

Agradezco a mis compañeros de laboratorio Ana Hernández Medina y Andrés Anzo, a mis hermanitos Roberto, Ernesto, Jessica y Héctor, así como a todos mis compañeros del IPICyT por los innumerables buenos ratos que pasamos juntos. De igual forma a la Dra. Imelda Bonifas por todas las risas y en especial por recibirme amablemente en su laboratorio durante todo este tiempo.

A mis compañeros de lucha Luis Javier, Ricardo Eliu, Gerardo y Jony, que aunque nos frecuentamos poco, esos momentos tan divertidos bien valen la pena.

Por último, pero lo más importante en mi vida, mi familia: a mi gran amor mi esposa, por el infinito tiempo que te he robado en la realización de esta tesis y con paciencia y cariño siempre me has impulsado a seguir adelante para lograr mis sueños, a mis padres “El Güero” y “Mari Tere” por darme siempre su apoyo incondicional sin importar la situación y en todos los ámbitos, a mis hermanos Emmanuel, Abraham y Diana por su apoyo. A mis sobrinas Andy y Sofi.

Aquellos amigos a quienes he omitido por descuido les pido perdón y de ante mano les agradezco su invaluable ayuda. Muchas gracias a todos.

Resumen

La seguridad de la información digital cada vez tiene más auge, debido al creciente uso de dispositivos móviles, además del incremento de operaciones realizadas a través de internet. Esto representa un gran reto ya que para lograr la confidencialidad, integridad y autentificación es necesario el uso de mecanismos especializados, una forma de proporcionar estos servicios es por medio de la criptografía, sin embargo, la creciente demanda requiere de nuevos algoritmos que sean más rápidos y a su vez más seguros. Una opción para lograr estos objetivos es la criptografía basada en sistemas caóticos.

Este trabajo de tesis se presenta en tres partes, en la primera parte se muestran los conceptos y definiciones básicas de las áreas de criptografía y sistemas dinámicos, así como las herramientas que se tienen para estudiar cada uno de estos sistemas. Además se da una visión general de estas áreas así como una clasificación, de tal forma que nos enfocaremos en cifrados en flujo y en sistemas dinámicos de tiempo discreto. Cabe señalar que los sistemas dinámicos estudiados en este trabajo a diferencia del mapeo logístico y casa de campaña son multi-modales, los cuales presentan ciertas ventajas en comparación con los mapeos uni-modales. Posteriormente se analizan las similitudes y diferencias que se encuentran en estas dos áreas, dando lugar a la criptografía caótica, la cual ha sido objeto de estudio por diversos grupos de investigación en los últimos años.

En la segunda parte de este trabajo se proponen metodologías para construir dos generadores pseudo-aleatorios, los cuales son la parte fundamental en el uso de cifrados en flujo, por un lado se propone el uso de valores positivos y negativos en el parámetro de bifurcación del mapeo logístico en conjunto con retardos, de esta forma es posible construir un generador que muestra resultados satisfactorios a las pruebas estadísticas de aleatoriedad propuestas por el NIST y además no es posible reconstruir el espacio fase. Por otro lado se presenta un generador basado en mapeos multi-modales el cual obtiene una secuencia binaria por medio de la combinación de diferentes modas, la principal ventaja radica en que solo es necesario definir un mapeo para obtener diferentes comportamien-

tos, de igual forma este generador presenta resultados satisfactorios al ser evaluado por las pruebas estadísticas propuestas por el NIST.

En la tercera parte de este trabajo se muestra el cifrado de imágenes en escala de grises por medio de diferentes funciones de cifrado, en cada caso se muestran los resultados de pruebas estadísticas realizadas para evaluar las propiedades del cifrado, en particular nos interesa el cifrado de imágenes, ya que los cifrados como el DES y el AES fueron diseñados para trabajar sobre cadenas de texto. Por lo tanto cifrar una imagen por medio de estos criptosistemas requiere de mayor capacidad de cómputo. Por último, estas funciones de cifrado son sometidas al ataque diferencial y de texto escogido en donde se muestra que propiedades deben de cumplir las funciones de cifrado para resistir este tipo de ataques.

Finalmente se muestran las conclusiones de esta tesis así como trabajo a futuro. Además, en la sección de apéndices se muestra la implementación electrónica del mapeo bi-modal, una breve descripción de las pruebas de aleatoriedad propuestas por el NIST y una extensión de este trabajo en donde se propone un generador pseudo-aleatorio basado en sistemas disipativos inestables.

Palabras Clave: criptografía, comportamiento caótico, cifrado en flujo, NIST, llave simétrica, mapeos multi-modales, generador pseudo aleatorio.

Abstract

The security of digital information is increasingly due to increasing use of mobile devices, in addition to the increase in transactions conducted via Internet. This represents a great challenge in order to achieve confidentiality, integrity and authentication is necessary use specialized mechanisms, a way of providing these services is through cryptography, however, the growing demand requires new algorithms that are faster and safe. One option to achieve these goals is chaos based cryptography.

This thesis is presented in three parts, in the first part the basic concepts and definitions of the cryptography and dynamic systems as well as the tools to study each of these systems are shown. Furthermore an overview of these areas and a classification is given, we will focus on stream ciphers and discrete time dynamical systems. Note that the dynamical systems studied in this work unlike the logistic and tent map are multi-modal, which have certain advantages compared with uni-modal maps. Afterwards the similarities and differences found in these two areas are analyzed, resulting in chaotic cryptography, which has been studied by several research groups in recent years.

In the second part of this work we construct two pseudo-random generators, which are an essential part in the use of stream ciphers, in one hand we propose use positive and negative values in the bifurcation parameter of the logistic map together with delays, in this way it is possible to build a generator showing satisfactory results to statistical tests of randomness proposed by the NIST and it is not possible to reconstruct the phase space. On the other hand we present a generator based on multi-modal maps which produce a binary sequence obtained by combining different modal, the main advantage is that it is only necessary to define one map for different behaviors, similarly this generator presents satisfactory results when is evaluated by statistical tests proposed by the NIST.

In the third part of this work the grayscale image encryption is shown using different encryption functions, in each case the results of statistical tests performed to evaluate the properties of encryption, in particular we are interested in image encryption due the cip-

hers like DES and AES are designed to work on text, therefore encrypt an image through these cryptosystems requires more computing power. Finally, these encryption functions are subjected to differential and chosen text attack where it is shown that properties must meet the encryption function to withstand such attacks.

Finally the conclusions of this thesis and future work is shown. Furthermore, in the appendix section contains an electronic implementation of bi-modal map, a brief description of tests of randomness proposed by the NIST and an extension of this work in which a pseudo-random generator is proposed based on unstable dissipative systems.

Keywords: Cryptography, chaotic behavior, stream cipher, symmetric key, multi-modal maps, pseudo random generator.

Índice general

Índice de figuras

Índice de tablas

Lista de acrónimos

$\Re$	—	denota el conjunto de números reales.
$\mathbb{N}$	—	denota el conjunto de números enteros.
$\mathbb{Z}^+$	—	denota el conjunto de números enteros positivos.
mod n	—	denota la operación módulo n.
$x \in X$	—	denota que el elemento x en mienbro del conjunto X.
$[a,b]$	—	denota que $\{x \in \Re : a \leq x \leq b\}$.
$[a,b)$	—	denota que $\{x \in \Re : a \leq x < b\}$.
$\sum_{i=1}^{n} b_i$	—	denota la suma $b_1 + b_2 + b_3 + \cdots + b_n$.
$A \subset B$	—	denota A está contenido en B.
$\forall\, x$	—	denota para toda x.
$A \cap B$	—	denota la intersección de los conjuntos.
$A \cup B$	—	denota la unión de los conjuntos.
ln x	—	logaritmo natural de x.
$\lfloor x \rfloor$	—	denota la operación piso.
$\emptyset$	—	denota conjunto vacio.
$\mathcal{M}$	—	denota el espacio del texto plano.
$\mathcal{C}$	—	denota el espacio del texto cifrado.
$\mathcal{K}$	—	denota el espacio de llaves.
$\mathcal{E}, \mathcal{D}$	—	denota los algoritmos de cifrado y descifrado.
$\oplus$	—	denota operación XOR.
x_0	—	denota condición inicial.
x^*	—	denota punto fijo.
$f : A \to B$	—	denota que f es una regla que asigna a cada elemento de $a \in A$ un elemento $b \in B$.
f'	—	denota derivada de f.

Glosario

Autenticación: Proceso por el cual se determina la identidad de un usuario.

Cifrado en bloques: Los cifrados por bloques toman grupos de tamaño fijo del texto plano y producen un bloque de tamaño fijo de texto cifrado.

Cifrado en flujo: Proceso por el cual el texto plano se combina mediante la operación XOR, con una secuencia pseudo-aleatorio del mismo tamaño, para generar un texto cifrado.

Cifrado de Vernam: Es un cifrado en flujo en el que el texto cifrado se obtiene a partir de la combinación con una secuencia aleatoria.

Criptosistema: Es un sistema que toma información legible para convertirlo en información no legible y viceversa.

Criptoanálisis: Es el conjunto de procedimientos, procesos y métodos empleados para romper un algoritmo criptográfico, descifrar un texto cifrado o descubrir las claves empleadas para generarlo.

Criptoanálisis diferencial: Es un ataque de texto plano elegido, se basa en el análisis de la evolución de las diferencias de dos textos planos relacionados cuando son cifrados con la misma clave.

Criptografía asimétrica: Criptosistema que utiliza dos llaves, una para cifrar y otra para descifrar información.

Criptografía caótica: Criptosistema que basa su funcionamiento en sistemas dinámicos caóticos.

Criptografía simétrica: Criptosistema que utiliza una llave para cifrar y descifrar información.

Confidencialidad: Es la propiedad que garantiza que la información es accesible sólo para aquellos autorizados a tener acceso.

Confusión: Establece que la relación entre la clave y el texto cifrado sea tan compleja como sea posible.

Difusión: Establece que pequeños cambios en el texto plano producen grandes modificaciones en el texto cifrado.

Espacio de fase: Es una construcción matemática, que permite representar gráficamente los posibles estados donde un sistema dinámico puede evolucionar.

Espacio métrico: Es un par (X,d) donde X es un conjunto no vacío y d es una función real definida, llamada distancia o métrica.

Exponente de Lyapunov: Es una cantidad que caracteriza el grado de separación de dos trayectorias infinitesimalmente cercanas.

Integridad: Proceso que permite saber si un mensaje llega su destino completo y sin alteraciones.

Sistema dinámico: Es un sistema cuyo estado evoluciona con el tiempo.

Texto cifrado: Mensaje el cual por medio de transformaciones no tiene sentido.

Texto plano: Mensaje legible que se desea enviar.

Capítulo 1

Introducción

1.1. Antecedentes

Desde épocas antiguas ha existido la necesidad de ocultar mensajes a personas no deseadas por medio de mensajes ocultos o cifrados, en la actualidad el interés por cifrar información no solo se ha mantenido vigente sino que además ha evolucionado y se ha adaptado a la tecnología de nuestros tiempos, dando como resultado el interés por ocultar información digital (archivos computacionales) pero ha mantenido su objetivo principal; dejar un mensaje ilegible, con la posibilidad de regresar a su forma original.

En general el mensaje que se desea enviar se le denomina **texto plano**, el cual por medio de transformaciones se convierte en texto sin sentido al que se le denomina **texto cifrado**, es importante mencionar que para que esta transformación tenga sentido es necesario que este proceso requiera de una llave, de tal forma que solo cuando se aplica la llave correcta el proceso es reversible.

La criptografía tiene un largo camino a través de la historia. El hombre a través del tiempo ha propuesto un sinfín de ideas, esquemas y algoritmos para cifrar información, entre los cifrados históricos más importantes se puede encontrar el llamado **cifrado por sustitución**, el cual consiste básicamente en intercambiar una letra del alfabeto por otra, de tal forma que para cifrar el mensaje se requiere de una tabla de sustitución, donde esta tabla es la llave, para recuperar el mensaje se aplica nuevamente la misma tabla de sustitución al mensaje cifrado. Otro de los cifrados históricos importantes que podemos encontrar es el **cifrado César**, el cual era usado en la antigua Roma, en su tiempo beneficio al pueblo romano ya que permitía transmitir mensajes y estrategias militares de tal

forma que aunque los mensajes fueran interceptados, los enemigos no eran capaces de interpretarlos. Este cifrado consistía en un desplazamiento de tres letras, por lo que este cifrado puede ser una generalización del cifrado de sustitución, ya que la clave en este caso es la tabla de sustitución que se forma al desplazar tres letras hacia adelante el abecedario, por ejemplo la letra a, es sustituida por la letra d, la b por la e, etc, un problema que se tenía era con las letras x, y, z en este caso comenzaba nuevamente el alfabeto por lo que la x le correspondía la letra a. Posteriormente se modificó este cifrado para que el número de corrimientos fuera parte de la llave y de esta forma la tabla de sustitución fuera diferente cada que se escogía una nueva llave. El principal problema que presentan todos los cifrados basados en sustitución, es que tienen una debilidad estadística, esto es si en el *texto plano* la letra **a** es la que tiene mayor número de repeticiones, entonces en el texto cifrado la letra que le corresponde será la que aparece con mayor número de repeticiones, de tal forma que aplicando un ataque estadístico es posible reconstruir la tabla de sustitución y de esta forma recuperar el mensaje original.

Otro cifrado histórico de gran relevancia es el llamado **cifrado Vigener**, este cifrado fue propuesto en el siglo XVI y se basaba en una propuesta totalmente diferente a los de sustitución. En este cifrado las letras son representadas por un número donde $a = 1, b = 2, \ldots, z = 26$, además se tenía un mensaje de longitud m y la $llave$ podía ser una palabra o frase de tal forma que si la longitud de la $llave$ k era menor que $m, k < m$ simplemente se repetía la $llave$ hasta que fuera de la misma longitud o mayor $m \leq nk$, posteriormente para cifrar la información se aplicaba letra por letra una suma entre el *texto plano* y la $llave$ que recordemos en este caso es una frase o palabra, de tal forma que el texto cifrado se podía representar como $c = k + m$, mod 26, este cifrado fue el primero en utilizar formalmente operaciones modulares, así se podía asegurar que la suma de letras siempre sería menor que 26 y así tener la letra correspondiente.

Posteriormente, con la llegada de la Revolución Industrial, se crearon máquinas de rotores para cifrar mensajes, la primera máquina de este tipo de la que se tiene registro se le conoce como la **máquina de Hebern**, la cual poseía un rotor. Para los primeros años de 1900's se inventó la **máquina Enigma**, esta máquina electromecánica es tal vez la más famosa para cifrar mensajes debido a que fue usada por los Nazis en la segunda guerra mundial. Se basaba en los cifrados de sustitución y en la máquina de Hebern, de tal forma que cambiaba una letra por otra de forma mecánica por medio de tres rotores, la llave de esta máquina estaba dada por la posición inicial de los rotores y era capaz de producir 17,576 combinaciones. En 1935 la máquina contaba con 4 rotores y para esa

fecha los alemanes la tomaron para uso oficial y exclusivo militar, con las modificaciones que realizaron la máquina era capaz de producir 10,967,424 combinaciones por esta razón se decía que era indescifrable. Con un funcionamiento avanzado para su época, la máquina Enigma alemana fue la que dio inicio a las primeras computadoras que se utilizaron para cifrar y descifrar códigos, además de servir como base para cifrados modernos. Más información acerca de cifrados históricos se puede encontrar en [1, 2] y referencias ahí mencionadas.

En general los cifrados históricos basan su funcionamiento en letras incluso la máquina Enigma, sin embargo en 1917 Gilbert Vernam ingeniero de la compañía AT&T propuso el cifrado que es conocido como **cifrado Vernam** [3], en el cual se proponía algo completamente diferente y significó el comienzo de la era digital, este cifrado basaba su funcionamiento en símbolos que solo podían tener dos valores, es decir un alfabeto binario, de igual forma el *texto plano* y el *texto cifrado* debían ser binarios, el proceso de cifrado se realizaba por medio de la operación Booleana XOR (OR-exclusiva, su funcionamiento se muestra en la figura 2.4), la llave es dada por un conjunto de bits. Posteriormente Joseph Mauborgne propuso que la llave fuera una secuencia totalmente aleatoria y a este cifrado se le conoció como **one-time-pad**, este cifrado tuvo y sigue teniendo una influencia en los cifrados modernos ya que fue el primer y único cifrado para el que existe una demostración de seguridad perfecta, la cual fue propuesta por Claude Shannon [4], en la sección 2.1.1 mostraremos a detalle este cifrado así como sus implicaciones.

En 1972 en Estados Unidos la dependencia llamada Buro Nacional de Estándares (NBS por sus siglas en inglés) ahora conocida como Instituto Nacional de Estándares y Tecnología (NIST) lanzó una convocatoria para crear un cifrado y establecerlo como un estándar en Estados Unidos. La idea era encontrar un cifrado que fuera seguro y pudiera ser usado en una variedad de aplicaciones. En 1974 recibieron una propuesta realizada por un grupo de criptógrafos que trabajan para la compañía IBM, el algoritmo fue llamado **cifrado Lucifer**. Lucifer es una familia de cifrados desarrollados por Horst Feistel a finales de los 1960's y fue uno de los primeros cifrados en bloque que operó sobre información digital. Un cifrado en bloques toma un conjunto de caracteres y los cifra simultáneamente, en particular Lucifer trabajaba con bloques de 64 bits usando una llave de 128 bits. Después de ser analizado por agencias gubernamentales, las que a su vez propusieron algunos cambios, este cifrado fue bautizado bajo el nombre de **DES** (Data Encryption Standard). Uno de los cambios más significativos que tuvo este cifrado, fue que DES se diseñó específicamente para resistir ataques de criptoanálisis diferencial, cabe mencionar

que este tipo de ataque no era de conocimiento público hasta el año de 1990 [5]. Entre otros cambios también se redujo el tamaño de la llave de 128 a 56 bits.

Finalmente en 1977 la versión final del cifrado DES fue dada a conocer públicamente [6], en la cual se describía el funcionamiento completo del algoritmo, sin embargo algunos criterios de diseño como las cajas de sustitución, nunca fueron descritos. Originalmente el cifrado DES se concibió para ser el estándar durante 10 años hasta 1987, sin embargo fue hasta el año de 1999 cuando fue remplazado.

Por otra parte en 1997 el NIST lanzó una convocatoria para establecer un nuevo cifrado el cual llevaría el nombre de **AES** (Advanced Encryption Standard), a diferencia del cifrado DES sería seleccionado de un concurso abierto administrado por el NIST, entre los requerimientos que se marcaban en la convocatoria este nuevo cifrado debería de manejar bloques de 128 bits, soportar tres diferentes longitudes de llave: 128, 192 y 256 bits. Fue en 2001 que el **cifrado Rijndael** propuesto por dos criptógrafos belgas Joan Daemen y Vincent Rijmen, fue anunciado oficialmente como el nuevo estándar para Estados Unidos [7].

Por otro lado en 1976 Whitfield Diffie y Martin Hellman [8] presentaron una propuesta revolucionaria que se basaba en la siguiente idea: No es necesario que la llave que se utiliza para cifrar la información sea secreta, la parte crucial radica en quien recibe la información (usuario B) y la llave que utiliza para descifrar es la que debe de ser secreta. De tal forma que para poder realizar este sistema, el usuario B da a conocer una llave pública que es conocida por todos y solamente sirve para cifrar, además el usuario B posee otra llave que es secreta y solo sirve para descifrar aquella información que haya sido cifrada con la llave pública, por lo tanto la llave del usuario B consiste de dos partes, una llave pública k_{pub} y una llave privada k_{pr}. Este tipo de criptografía es conocida como **asimétrica**, actualmente existe una gran variedad de cifrados basados en esta idea, sin embargo los más usados son el **cifrado RSA** [9] que se basa en el problema de factorización de números enteros, el **cifrado Elgamal** [10] que se basa en el problema del logaritmo discreto, por último, se tiene el cifrado basado en **curvas elípticas** [11], el cual se basa en el problema del logaritmo discreto pero se aborda desde otro punto de vista. Principalmente los cifrados RSA y Elgamal son computacionalmente intensos, por lo que resulta impráctico tratar de cifrar una gran cantidad de información utilizando estos algoritmos, por otro lado los criptosistemas basados en curvas elípticas tienen considerablemente un mejor desempeño; sin embargo, no pueden competir en velocidad con los cifrados simétricos como el AES. Los sistemas de llave pública nos brindan la posibilidad de resolver problemas como

la integridad y la autenticidad además de la confidencialidad, por esta razón actualmente se emplean en conjunto cifrados simétricos y asimétricos.

Todos los cifrados que se han mencionado hasta ahora se clasifican como cifrados convencionales, existe una gran variedad de cifrados propuestos los cuales se basan en diferentes procesos, por ejemplo en autómatas celulares [12], montajes ópticos [13], etc. Una propuesta muy prometedora que ha crecido en los últimos años es la de criptosistemas basados en sistemas dinámicos caóticos, debido a que poseen propiedades que son análogas en el campo de la criptografía, a este campo se le ha denominado como **criptografía caótica** la cual se mostrara a detalle en la Sección 2.3.

La teoría de los sistemas dinámicos caóticos comenzó a desarrollarse en 1960, siguiendo el texto de Strogatz [14], uno de los primeros trabajos que marcaron el inicio de los sistemas caóticos fue realizado por Henri Poincaré, con su trabajo Ciencia y Método [15] el cual dio un giro a la forma en la que se analizaban los sistemas, el mencionó que cuando un sistema determinista presenta sensibilidad a las condiciones iniciales es imposible realizar una predicción de la evolución del sistema a largo plazo. Sin embargo el estudio de estos sistemas quedó ignorado la primera mitad del siglo XX y en su lugar se desarrolló el estudio de osciladores no lineales, con esto se propusieron nuevas técnicas matemáticas para el análisis.

Con el desarrollo de las computadoras en la década de los 50's fue posible experimentar con ecuaciones una forma que antes era imposible, gracias a esto Lorenz pudo desarrollar su trabajo en 1963 [16], estudiaba un modelo climático simplificado de tres ecuaciones diferenciales y encontró que las soluciones de su sistema nunca se establecían en un punto de equilibrio o en una solución periódica. Además si comenzaba su simulación con condiciones iniciales muy cercanas estas rápidamente arrojaban resultados totalmente diferentes. El trabajo de Lorenz tuvo difusión hasta la década de los 70's en la cual se desarrolló de forma amplia la teoría del caos, podemos destacar algunos trabajos como el de Metrópolis y colaboradores [17] en el cual caracterizan el comportamiento de aplicaciones uni-dimensionales utilizando por primera vez la dinámica simbólica. El trabajo de Li y Yorke [18] utilizan la palabra caos para describir el comportamiento de diversos sistemas deterministas y destacan que si un sistema exhibe órbitas de periodo tres, puede exhibir periodos de cualquier tamaño. El trabajo de Robert May [19] demostró que un sistema discreto de una dimensión basado en un modelo de crecimiento de población presenta un comportamiento irregular e impredecible y destacó la importancia de estudiar sistemas no lineales simples. Grossmann y Thomae [20] caracterizaron la distribución es-

tadística de sistemas discretos uni-dimensionales y demostraron que es invariante. Mandelbrot propuso el término fractal y produjo gráficas auto semejantes en diferentes escalas [21, 22]. Feigenbaum descubrió ciertas leyes universales que gobernaban la transición de un comportamiento regular a uno caótico [23]. La primera implementación electrónica de un sistema caótico fue propuesta por Chua [24], posteriormente en el trabajo de Percora y Carrol [25] se mostró bajo qué condiciones es posible sincronizar estos circuitos.

En la literatura se puede encontrar gran variedad de referencias en las que se estudian a fondo los sistemas dinámicos de tiempo discreto, entre las cuales podemos encontrar [26, 27, 28], los sistemas uni-modales que se les conoce como sistemas clásicos, en específico el mapeo logístico se ha estudiado para valores positivos en el parámetro de bifurcación, sin embargo en [29] se muestra que el parámetro de bifurcación puede tomar valores negativos. Por otro lado en [30] se presentaron los mapeos multi-modales los cuales presentan ciertas ventajas respecto a los mapeos uni-modales, sin embargo los autores no mostraron que este tipo de sistemas presenten comportamiento caótico, en la sección 2.2.2 se muestra lo referente a mapeos caóticos.

Los primeros cifrados que relacionan a los sistemas caóticos con la criptografía se presentaron como una posible aplicación de la teoría del caos [31, 32, 33]. En este primer acercamiento se explotaba el comportamiento aparentemente aleatorio de las órbitas generadas al evolucionar el sistema caótico, por lo que eran utilizados como generadores pseudo-aleatorios, una parte de esta tesis se enfoca en proponer dos generadores pseduo-aleatorios. A partir de entonces comenzaron a construirse una gran variedad de cifrados basados en sistemas caóticos, estos se pueden clasificar en dos tipos: aquellos que se basan en sistemas continuos y los que se basan en sistemas discretos, en este trabajo de tesis nos enfocaremos en los sistemas de tiempo discreto. En la sección 2.3 se atenderá en detalle cada uno de estos criptosistemas.

1.2. Estado del arte

Por un lado en este trabajo de tesis se proponen dos diferentes generadores pseudo-aleatorios basados en sistemas dinámicos discretos que presentan comportamiento caótico y posteriormente pueden ser usados en cifrados en flujo, a continuación se muestra el estado del arte de esta rama en específico.

González y colaboradores [34, 35] propusieron una función basada en el mapeo logístico la cual produce secuencias aleatorias o pseudo-aleatorias, de igual forma Stojanovski

y Kocarev [36, 37] propusieron un generador aleatorio de bits basado en un mapeo lineal por partes, estos generadores al ser completamente aleatorios no existe forma de recuperar o reconstruir la secuencia a partir de una condición inicial, sin embargo estos algoritmos pueden ser útiles para la generación de llaves.

Andrecut [38] propuso un algoritmo para obtener secuencias pseudo-aleatorias basadas en el mapa logístico, donde mencionaba que este generador es aperiódico, infinito y sin correlación, sin embargo esta es una idealización del comportamiento del mapeo logístico ya que al implementar cualquier mapeo en una máquina de precisión finita se tendrán órbitas periódicas. Algunos trabajos como el de Liu [39] y el de Wang [40] proponen diferentes algoritmos para obtener secuencias pseudo-aleatorias de bits basadas en un solo mapeo caótico. Sin embargo Shujun Li y colaboradores mostraron en [41] que los generadores pseudo-aleatorios basados en un solo mapeo caótico son potencialmente inseguros ya que puede existir información acerca del sistema caótico, además proponen un generador pseudo-aleatorio basado en dos mapeos caóticos, los cuales son iterados de forma independiente y la salida está dada por un algoritmo de comparación entre los dos sistemas.

Tomando como base el trabajo de Shujun Li [41] su puede encontrar una gran variedad de generadores pseudo-aleatorios basados en la mezcla de varios sistemas caóticos. En el trabajo de Kanso A. [42] se muestran varios generadores basados en la mezcla de dos mapeos logísticos, sin embargo no se reporta un análisis estadístico, Patidar V. y colaboradores en [43] iteran de forma independiente dos mapeos logísticos y en base a ellos construyen una secuencia de bits, posteriormente presentan un análisis estadístico, bajo la misma idea proponen un algoritmo basado en un mapeo bi-dimensional [44]. En [45] los autores proponen una modificación al mapeo logístico de forma que al iterar el sistema se alterna la aplicación del sistema caótico y de esta forma emulan el uso de dos sistemas caóticos. En [46] proponen un algoritmo en el cual por medio de un mapeo caótico inducen una perturbación a otro mapeo, con esto garantizan aumentar la longitud de las órbitas ya que en cada iteración cambia el valor que naturalmente se tendría en el mapeo. Recientemente François y colaboradores [47] presentaron un generador basado en la mezcla de tres mapeos caóticos iterados de forma independiente. Se puede observar que la gran mayoría de los generadores propuestos están basados en el mapeo logístico y en general en mapeos uni-modales.

Este trabajo de tesis se enfoca en **el diseño y evaluación de generadores pseudo-aleatorios** basados en sistemas caóticos de tiempo **discreto** así como su aplicación a

cifrados en flujo, de forma general se puede dividir en dos esta propuesta. Por un lado, se propone un generador basado en el mapeo logístico, para este generador se toman en cuenta las propiedades que se reportaron en [29] en donde se muestra que el parámetro de bifurcación del mapeo puede tomar valores negativos, con esto se generan dos series de tiempo independientes, además para aumentar la complejidad de las series estas son retroalimentadas usando retardos, tanto en la serie con parámetro de bifurcación positivo como en la serie con parámetro de bifurcación negativo. Por otro lado en [30] se propuso una familia de mapeos multi-modales basados en la forma del mapeo logístico, estos mapeos presentan ventajas frente a los clásicos mapeos uni-modales, ya que tienen un amplio rango de valores validos del parámetro de bifurcación, además variando este parámetro pueden cambiar su comportamiento sin necesidad de cambiar su función, es decir, con la variación de un parámetro se tiene una variación en el número de modas, sin embargo no se mostró si estos sistemas tienen comportamiento caótico. Tomando en cuenta las propiedades de estos sistemas se propone un generador basado en la mezcla de series de tiempo de mapeos **multi-modales**.

1.3. Definición del problema

La criptografía clásica se basa en el uso de algoritmos, en la teoría de números y en algunos casos como los cifrados en flujo se basa en comportamiento similar al ruido, los sistemas dinámicos con comportamiento caótico producen secuencias que se asemejan al ruido sin embargo su origen proviene de un comportamiento determinista el cual es sensible a pequeñas variaciones en las condiciones iniciales o parámetros.

Por medio de los sistemas dinámicos es posible construir generadores pseudo-aleatorios, tal como se mostró en el estado del arte, además, se pudo observar que se utilizan sistemas uni-modales. La hipótesis bajo la cual se realiza este trabajo consiste en que el uso de mapeos **multi-modales** en generadores pseudo-aleatorios ofrece mejores condiciones comparado con los mapeos uni-modales para obtener secuencias binarias que poseen propiedades estadísticas similares a las que se observan en secuencias binarias aleatorias.

1.4. Objetivo general

Desarrollar un generador pseudo-aleatorio basado en mapeos multi-modales que presente propiedades estadísticas semejantes a las que presentan las secuencias aleatorias, además utilizar estas secuencias en cifrado en flujo con imágenes en escala de grises y comparar los resultados con diferente número de modas, para evaluar las propiedades se utilizaran las pruebas estadísticas de aleatoriedad del NIST, entropía, correlación de pixeles y calidad de cifrado.

1.4.1. Objetivos particulares

- Analizar y caracterizar mapeos caóticos discretos uni-modales así como la obtención de diagramas de bifurcación.
- Generar secuencias binarias pseudo-aleatorias.
- Generar diagramas de bifurcación de mapeos multi-modales.
- Diseñar metodologías para producir secuencias pseudo- aleatorias con un número deseado de modas.
- Realizar pruebas de aleatoriedad en las secuencias generadas.
- Diseñar e implementar prototipos de sistemas de cifrado de imágenes.
- Realizar pruebas de seguridad a los sistemas de cifrado.

1.5. Justificación

El área de seguridad informática se encuentra en constante desarrollo ya que continuamente se requieren de mayores exigencias para garantizar que la información se mantenga integra y confidencial, es por ello que continuamente se buscan alternativas en el diseño y desarrollo de cifrados.

El uso de sistemas dinámicos es una alternativa en el diseño y desarrollo de procesos de cifrado, ampliando así, el panorama en el área.

1.6. Organización

En el capítulo dos se presentan los conceptos y definiciones necesarias para este trabajo de tesis, este capítulo se divide en tres partes, en la primera sección se presentan los conceptos de la criptografía así como su clasificación centrándonos en los cifrados en flujo, como se mostrará el núcleo de estos cifrados son los generadores pseudo-aleatorios por lo que se dará una definición de los distintos tipos de generadores, por último se mostraran las bases del criptoanálisis.

En la segunda sección de este capítulo se presentan los conceptos de sistemas dinámicos y su clasificación en sistemas de tiempo continuo o discreto, así como las propiedades que se requieren para que un sistema dinámico presente comportamiento caótico, nos enfocaremos en los sistemas caóticos de tiempo discreto y se mostraran desde mapeos uni-modales hasta multi-modales, además se abordarán las herramientas que son necesarias para el estudio de estos sistemas como lo son: puntos fijos, diagramas de bifurcación, exponente de Lyapunov, etc.

En la tercera sección de este capítulo se mezclan las áreas de criptografía y sistemas dinámicos dando como resultado la criptografía caótica, se mostrarán las similitudes y diferencias entre estas áreas y se mostrarán criptosistemas basados en sistemas caóticos.

En el capítulo tres se muestran dos generadores pseudo-aleatorios uno de ellos basado en el mapeo logístico y el otro en mapeos multi-modales, estos generadores son analizados por medio de pruebas estadísticas y se muestra que son seguros para su uso en el área de la criptografía.

En el capítulo cuatro se usan los generadores diseñados en el capítulo tres y se llevan a la aplicación del cifrado de imágenes, este capítulo se divide en cuatro partes. En la primera sección se muestra el banco de pruebas estadísticas con las que se evaluarán las funciones de cifrado, es importante remarcar que estas pruebas son diferentes a las que se usaron en el capítulo tres. En la segunda sección se implementa la función de cifrado más básica, la cual consta de una operación XOR, esta función es evaluada y se muestran sus ventajas así como sus puntos débiles. En la tercera sección se presenta una función de cifrado la cual utiliza retardos, por medio del análisis estadístico se muestran sus puntos débiles y sus ventajas comparándolo con la función de cifrado anterior. Por último en la cuarta sección se presenta una función de cifrado que incluye nuevos elementos.

Finalmente en el capítulo cinco se muestran las conclusiones que se obtuvieron en este trabajo de tesis.

Capítulo 2

Conceptos y definiciones

2.1. Criptografía

La criptografía se enfoca en resolver los problemas de confidencialidad, integridad y autenticación [48] por medio de una variedad de herramientas, el término general con el que se engloba el área de criptografía se denomina criptología, ver figura 2.1, la cual se divide en dos ramas (La parte correspondiente al criptoanálisis se muestra en la figura 2.7).

Una definición de **criptografía** es, la ciencia de la escritura secreta cuyo objetivo es ocultar el significado de un mensaje. Por otro lado tenemos **criptoanálisis** definido como la ciencia y algunas veces el arte de romper criptosistemas.

Existen varios tipos de cifrados, por un lado los **cifrados simétricos** permiten a dos usuarios compartir información cifrada haciendo uso de dos algoritmos uno para cifrar y

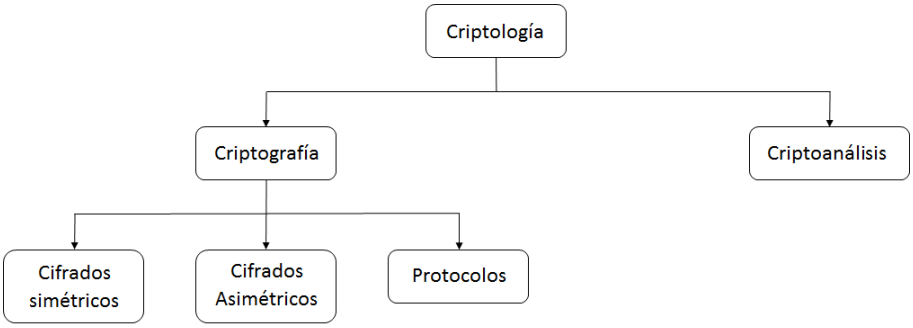


Figura 2.1: Esquema general de la criptología.

otro para descifrar, la criptografía desarrollada desde tiempos remotos hasta 1976 estaba basada en métodos simétricos, en donde se utiliza la misma llave para cifrar y descifrar, este tipo de cifrados aún siguen en uso y son objeto de investigación y desarrollo. **Con los cifrados simétricos se puede resolver el problema de confidencialidad**. Existen dos clases dentro de esta clasificación, los cifrados en flujo y los cifrados en bloque.

Los **cifrados en bloque** toman un conjunto de caracteres de longitud fija (típicamente la longitud es de 64, 128 o 256 bits) y los cifran simultáneamente. Cuanto mayor sea el tamaño del bloque, más seguro será el sistema de cifrado con la desventaja de que aumenta la complejidad del algoritmo haciéndolo más lento. Es importante mencionar que los cifrados en bloque pueden comportarse como generadores pseudo-aleatorios, de esta forma los cifrados en bloque pueden ser evaluados por medio de pruebas estadísticas.

Los **cifrados en flujo** toman la información bit a bit por lo que son mucho más rápidos que un cifrado en bloque. El principal componente de estos sistemas es un generador pseudo-aleatorio (PRNG por sus siglas en inglés), donde el texto cifrado se obtiene a partir de la combinación de la secuencia generada con la información por medio de la operación O-exclusiva (XOR).

Los **cifrados asimétricos** fueron propuestos en 1976 por Whitfield Diffie y Martin Hellman [8], estos basaban su funcionamiento en una propuesta totalmente diferente a lo que se había manejado. En la criptografía asimétrica se tiene una llave privada al igual que en los cifrados simétricos pero se incluye el uso de una llave pública. Con estos sistemas se puede logar el intercambio de llaves, el cifrado de información y firmas digitales las cuales proveen la autenticación.

Los **protocolos** son algoritmos que proveen principalmente la integridad, comúnmente se usan en conjunto con cifrados simétricos y asimétricos, entre ellos se encuentran las funciones Hash y los códigos de autenticación de mensaje (MACs por sus siglas en inglés).

Con base en los conceptos anteriores podemos definir formalmente un cifrado simétrico como:

Definición 1. *Un criptosistema simétrico es un quinteto* $(\mathcal{M}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ *que satisface las siguientes condiciones [49]*

- $\mathcal{M}$ *denota al espacio del texto plano o mensaje, esto es, es el conjunto finito de todos los textos planos posibles.*

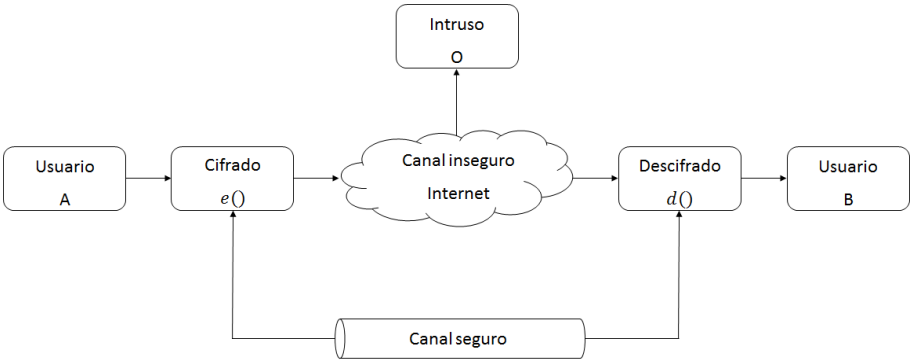


Figura 2.2: Criptosistema de llave simétrica.

- *$\mathcal{C}$ es el espacio del texto cifrado, es decir, es el conjunto finito de todos los textos cifrados posibles.*

- *$\mathcal{K}$ es el conjunto finito de todas las posibles llaves, también llamado espacio de llaves.*

- *$\mathcal{E}, \mathcal{D}$, conjunto de posibles reglas, esquemas o algoritmos de cifrado y descifrado. Para cada $k \in \mathcal{K}$, existe un proceso de cifrado $e_k \in \mathcal{E}$ y un proceso de descifrado correspondiente $d_k \in \mathcal{D}$ de manera que, $d_k(e_k(m)) = m$ para cada texto plano $m \in \mathcal{M}$.*

De esta forma un usuario A envía la información cifrada por medio de un canal inseguro y en caso de existir algún intruso, lo único que este podría obtener es texto sin sentido pero el usuario B tiene posibilidad de obtener la información correcta ver figura 2.2.

Para construir un criptosistema de forma correcta se deben de tomar en cuenta los siguientes aspectos:

- Los fundamentos de la teoría de la información descritos por Shannon [4, 50] donde describe que un buen criptosistema exhibe las propiedades de confusión y difusión, con esto se logra que el texto cifrado sea estadísticamente independiente de la llave y del texto plano. La propiedad de difusión nos dice que pequeños cambios en el texto plano producen grandes modificaciones en el texto cifrado. En los cifrados por bloques esto se realiza por medio de permutaciones. Por otro lado la confusión señala que la relación entre la clave y el texto cifrado sea tan compleja como sea

posible, típicamente en los cifrados por bloque esto se logra por medio de las cajas de sustitución. Por lo tanto un criptosistema debe de ser sensible al texto plano, sensible a la llave y no debe de existir ningún patrón que relacione el texto plano con el texto cifrado.

- Tener un gran espacio de llaves de forma que sea ineficiente un ataque de fuerza bruta en donde se prueba una a una la posible llave del sistema hasta encontrar la correcta.

Además de lo anterior un criptosistema sólido debe de cumplir el principio de Kerckhoffs que establece lo siguiente [51]. Un criptosistema debe de ser seguro incluso si el atacante (intruso) conoce todos los detalles acerca del sistema, con la excepción de la llave secreta. En particular el sistema debe ser seguro cuando el atacante conoce los algoritmos de cifrado y descifrado.

2.1.1. Cifrado en flujo

En este trabajo de tesis el principal esquema a investigar son los cifrados en flujo por esta razón entraremos en detalles de este tipo de sistemas.

Los cifrados que se desarrollaron a lo largo de la historia basaban su funcionamiento en letras, por esta razón los algoritmos de cifrado utilizaban operaciones modulares. El primer cifrado que basó su funcionamiento en un sistema binario fue propuesto por Gilbert Vernam [3] este cifrado también se le conoce como “One-Time-Pad”, en el cual se toma el texto plano (x_i), el texto cifrado (y_i) y una secuencia de caracteres binarios (s_i) de forma individual, es decir bit a bit. Las funciones para cifrar y descifrar se muestran a continuación:

$$\begin{aligned} y_i &= e_{s_i} = x_i + s_i, \mod 2; \\ x_i &= d_{s_i} = y_i + s_i, \mod 2. \end{aligned} \tag{2.1}$$

En la figura 2.3 se muestra un esquema general de este cifrado que a pesar de parecer bastante simple tiene grandes implicaciones. La primera de ellas es que la función para cifrar y descifrar son muy similares, sin embargo son inversas, es decir si aplicamos la operación $d_{s_i}(y_i) = x_i$, como se muestra a continuación:

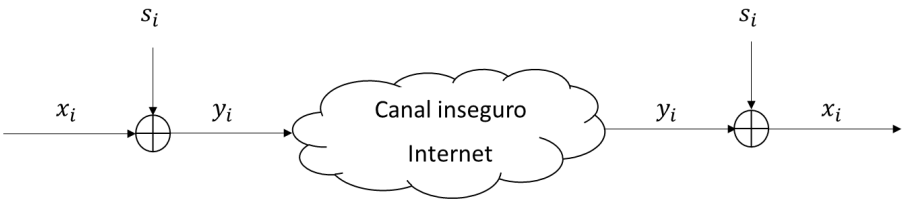


Figura 2.3: Esquema general del cifrado Vernam.

x_i	s_i	$y_i = x_i + s_i \bmod 2$
0	0	0
0	1	1
1	0	1
1	1	0

Figura 2.4: Operación Booleana XOR.

$$\begin{aligned} d_{s_i}(y_i) &= y_i + s_i && \bmod 2 \\ d_{s_i}(y_i) &= (x_i + s_i) + s_i && \bmod 2 \\ d_{s_i}(y_i) &= x_i + 2s_i && \bmod 2 \\ d_{s_i}(y_i) &= x_i + 0 && \bmod 2 \\ d_{s_i}(y_i) &= x_i && \bmod 2 \end{aligned} \tag{2.2}$$

Otra implicación importante de este cifrado es que al realizar operaciones módulo 2, los únicos posibles valores resultantes son 0 y 1, por lo que podemos manipular esta operación módulo 2 como una operación Booleana y en específico es equivalente a la operación XOR como se muestra en la figura 2.4.

Shannon [4] demostró matemáticamente que el texto plano nunca podría deducirse a partir del texto cifrado, incluso con recursos computacionales infinitos, esto es, la información que aporta el texto cifrado acerca del mensaje es nula, a esto le llamo seguridad incondicional y se da cuando el cifrado de Vernam se utiliza bajo las siguientes condiciones:

1. La secuencia de bits $s_0, s_1, s_2, \ldots$ se obtiene por medio de un generador aleatorio.
2. La secuencia s_i solo es conocida por los usuarios legítimos A y B.
3. La secuencia s_i es usada solamente una vez.

La primera condición nos dice que se requiere de un generador aleatorio, esto se logra por medio de un dispositivo como un semiconductor o un generador basado en ruido. La segunda condición nos dice que la secuencia generada debe ser compartida por medio de un canal seguro y la última condición nos marca que la secuencia no puede ser usada más de una vez. Esto implica que por cada bit de información requerimos un bit de la secuencia, por lo tanto la llave del sistema (secuencia generada) será del mismo tamaño que la información a transmitir. Esto hace que sea impráctico llevar este sistema a su aplicación en la realidad, sin embargo nos da una gran idea para el diseño de cifrados seguros. Los cifrados en flujo toman la idea del cifrado Vernam y lo mejoran, para esto utilizan otra forma de generar la secuencia por lo que necesitamos de los siguientes conceptos [52].

Definición 2. *Generador de números aleatorios (TRNG): se caracteriza por el hecho de que su salida no puede ser reproducida. Este tipo de generador se basa en procesos físicos como el ruido de semiconductores.*

Definición 3. *Generador de números pseudo-aleatorios (PRNG): las secuencias son generadas a partir de una condición inicial o una semilla, estas secuencias no son aleatorias ya que se obtienen por procesos totalmente deterministas. Un requerimiento de estos generadores es que tengan buenas propiedades estadísticas, esto significa que las secuencias generadas deben aproximarse a secuencias aleatorias.*

Definición 4. *Generador de números pseudo-aleatorios criptográficamente seguro (CSPRNG): es un tipo especial de PRNG que posee una propiedad adicional, es impredecible. Esto significa que dados n bits consecutivos de la secuencia generada, no existe algún algoritmo que pueda predecir el siguiente bit s_{n+1} con una probabilidad más alta del 50%. Por otra parte debe de ser computacionalmente inviable calcular cualquier bit anterior s_{n-1}, s_{n-2}.*

Los cifrados en flujo reemplazan el generador aleatorio (TRNG) por un generador pseudo-aleatorio criptográficamente seguro (CSPRNG) de forma que se reduce el problema de transmitir de forma segura una secuencia larga por una semilla o una condición inicial del generador. Ver figura 2.5.

Es importante mencionar que el concepto de seguridad incondicional descrito por Shannon requiere que la llave (secuencia generada) sea de la misma longitud que el texto plano, por lo tanto todos los criptosistemas modernos no son incondicionalmente seguros, para estos existe otra clasificación que nos indica que son computacionalmente seguros.

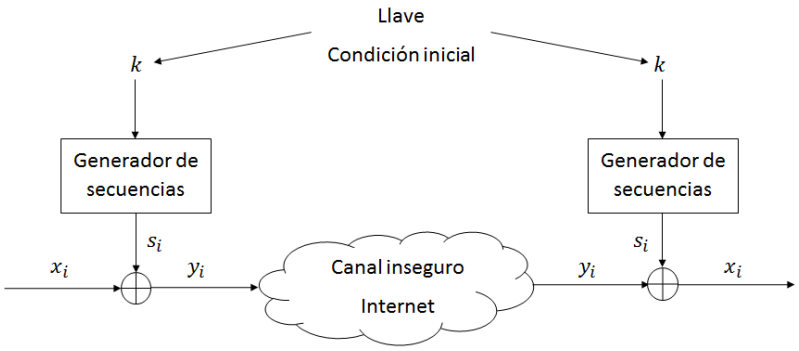


Figura 2.5: Esquema general del cifrado en flujo.

Definición 5. *Un criptosistema es computacionalmente seguro si el mejor algoritmo para romper el sistema requiere al menos t operaciones.*

Por otro lado podemos observar que el núcleo de los cifrados en flujo es el generador por lo que la seguridad del cifrado radica en la generación de la secuencia pseudo-aleatoria, sin embargo no existe un criterio general que determine con exactitud la seguridad de un generador pseudo-aleatorio, no obstante es posible señalar ciertas propiedades que toda secuencia debe de cumplir para su uso en los cifrados en flujo, entre las características necesarias pero no suficientes se encuentra la distribución. Típicamente esta y otras características se evalúan comparando la secuencia generada con una secuencia realmente aleatoria utilizando una serie de pruebas estadísticas. Las pruebas utilizadas en esta tesis se describen en el apéndice B.

Por último existen dos modos de operación para los cifrados en flujo: síncronos y asíncronos. Para el caso de los síncronos estos dependen solamente de la llave mientras que para los asíncronos dependen tanto de la llave como del texto cifrado por medio de una retroalimentación como se muestra en la figura 2.6

La mayoría de los cifrados en flujo trabajan de forma síncrona sin embargo existen algunos cifrados asíncronos como los de registro de desplazamiento con retroalimentación lineal (LFSR) los cuales tienen una distribución estadística que se asemeja a una secuencia aleatoria sin embargo basta con conocer el doble de la longitud de la secuencia para poder construir el generador.

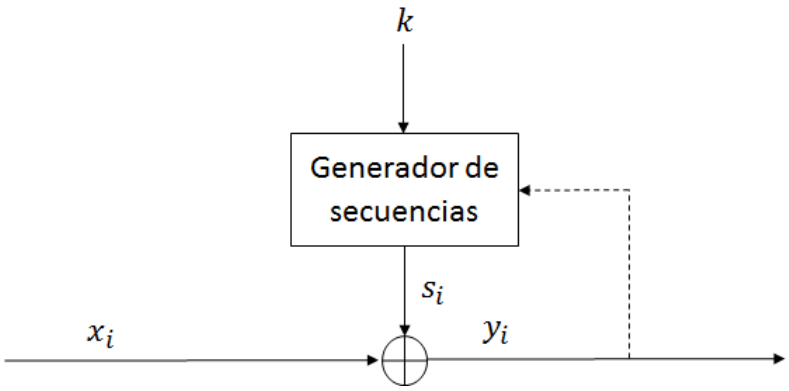


Figura 2.6: Cifrado en flujo síncrono y asíncrono.

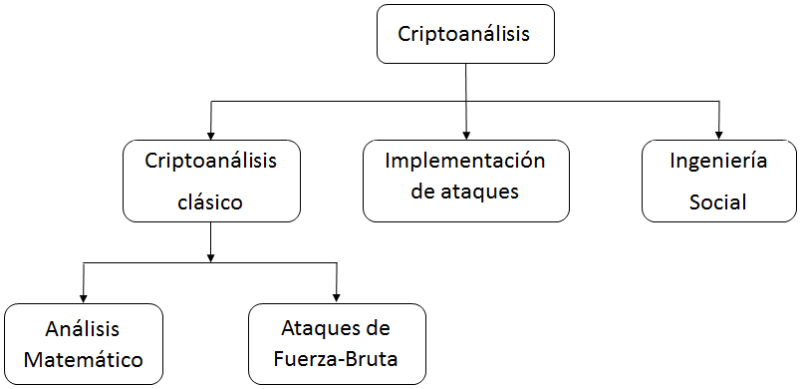


Figura 2.7: Esquema general del criptoanálisis.

2.1.2. Bases del criptoanálisis

Existe una gran variedad de ataques y formas de romper cifrados, estas no están categorizadas, sin embargo podemos ver de forma general la clasificación del criptoanálisis como se muestra en la figura 2.7.

El **criptoanálisis clásico** es la ciencia que se encarga de recuperar el texto plano *x* a partir del texto cifrado *y* o de forma alternativa recuperar la llave *k*. Se puede dividir en dos ramas que son **ataques analíticos** y **ataques de fuerza bruta**, los primeros explotan la estructura interna del algoritmo de cifrado. De acuerdo a [49], existen diferentes niveles de ataques:

1. Ataque al texto cifrado.

2. Ataque por texto plano conocido.

3. Ataque por texto plano elegido.

4. Ataque por texto cifrado elegido.

Por otro lado los ataques de **fuerza bruta** toman el algoritmo como una caja negra y consiste en probar todas las posibles claves. En la actualidad un criptosistema es vulnerable a un ataque de fuerza bruta cuando la longitud de su clave es inferior a 112 bits [53, 54]

La **implementación de ataques** tiene lugar en aquellos criptosistemas en donde el atacante tiene acceso físico y trata de obtener la llave por medio de la medición de alguna señal eléctrica, radiación electromagnética, el tiempo de procesamiento de algoritmos así como el consumo de potencia.

Los ataques de **ingeniería social** involucran a humanos para obtener la llave secreta, por ejemplo cuando alguien es forzado para obtener la clave por medio de chantajes, engaños, amenazas o en su caso sobornos.

Un criptosistema sólido además debe adherirse al principio de Kerckhoffs [51] y mantener abiertos al público todos los detalles del sistema excepto la llave, de otra forma se consigue la seguridad mediante la obscuridad. A lo largo de la historia se ha demostrado que los sistemas que consiguen seguridad por obscuridad a menudo son débiles y se pueden romper fácilmente usando ingeniería inversa, un ejemplo reciente de este hecho es el algoritmo de cifrado para la protección del contenido de películas en formato DVD, tiene como nombre “Content Scrambling System (CSS)” el cual se rompió con facilidad utilizando ingeniería inversa [55].

2.2. Sistemas dinámicos

Un sistema dinámico determinista puede ser visto como un conjunto de operadores o reglas de evolución F^t que determinan un campo vectorial que da la dinámica del estado del sistema x_t en el tiempo $t \in T$, con una condición inicial conocida x_0. Este conjunto de reglas actúa en algún espacio de estados o espacio de fase, el cual es un espacio métrico. La forma más general para especificar la evolución del sistema es asumiendo que para $t \in T$ el mapa F^t está definido en el espacio de estados $X \subset \mathbb{R}^n$ siendo n su dimensión.

$$F^t : X \to X$$

el cual transforma un estado inicial x_0 en otro estado $x_t \in X$ en el tiempo t:

$$x_t = F^t x_0.$$

El campo vectorial tiene dos propiedades naturales las cuales reflejan la propiedad determinista de un sistema dinámico:

$\mathbb{A}$. $F^0 = id$, donde id es la identidad del mapa sobre X, $idx = x$ para todo $x \in X$. Esta propiedad implica que el sistema no cambia su estado espontáneamente.

$\mathbb{B}$. $F^{t+s} = F^t \circ F^s$, significa que $F^{t+s} = F^t(F^s x),\ \forall\ x \in X$ y $t,s \in T$ tal que ambos lados de la ecuación están definidos. Esencialmente esta propiedad establece que el campo vectorial del sistema no varía en el tiempo, con esto se dice que el sistema es autónomo.

Definición 6. *Un sistema dinámico determinista es una terna $\{T,X,F^t\}$, donde T es el tiempo, X es el espacio de estados y $F^t : X \to X$, es una familia de operadores de evolución parametrizados por $t \in T$ que satisfacen las propiedades $\mathbb{A}$ y $\mathbb{B}$ [56].*

Para facilitar la compresión y las propiedades de estos sistemas se utilizan objetos geométricos. Los objetos geométricos básicos asociados a un sistema dinámico $\{T,X,F^t\}$ son sus órbitas en el espacio de estado.

Definición 7. *Una órbita que comienza en x_0 es un conjunto ordenado del espacio de estados X.*

$$Or(x_0) = \{x \in X : x = F^t x_0,\ \forall\ t \in T\} \textit{tal que } F^t x_0 \textit{ está definido}.$$

Las órbitas más simples son las que se mantienen en equilibrio, también se les conoce como órbitas estacionarias o invariantes.

Definición 8. *Un punto $x^* \in X$ es llamado punto fijo de un sistema dinámico determinista si $x(t) = x^*\ \forall\ t \in T$.*

Otro tipo de órbita relativamente simple es llamada periódica.

Definición 9. *Una órbita es periódica L_0, si para cada punto $x(t) \in L_0$ satisface que $x(t+T_0) = x(t)$ para algún $T_0 > 0\ \forall\ t \in T$. El T_0 mínimo que cumpla con esta condición se denomina el periodo fundamental de la órbita L_0.*

Aquí se pueden distinguir dos casos: sistemas dinámicos continuos y sistemas dinámicos discretos. Si la evolución del sistema respecto al tiempo se aplica de la forma $T \in \Re$

el sistema dinámico es de tiempo continuo y se describe por medio de ecuaciones diferenciales.

$$\dot{x}(t) = F(x(t)) \tag{2.3}$$

donde $\dot{x} = \frac{d}{dt}x(t), x \in \Re$. $F : \Re^n \to \Re^n$. El vector $x(t)$ representa el vector de estado del sistema y está formado por $x(t) = (x_1, x_2, \ldots, x_n)$ que son llamadas variables de estado.

Por otra parte si la evolución del sistema respecto al tiempo se aplica de la forma $T \in \mathbb{N}$ el sistema dinámico es de tiempo discreto y se describe por medio de ecuaciones en diferencias.

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

donde $n \in \mathbb{Z}^+, x_n \in X \subset \Re$, $F : X \to X$. Las órbitas son el conjunto de puntos de la secuencia, son numeradas por enteros crecientes, para este tipo de sistemas la órbita más simple es dada cuando el sistema evoluciona a un punto fijo. Se puede describir la relación entre un sistema dinámico continuo y uno discreto a partir del mapeo de Poincaré con el cual se reduce la dimensión del sistema a $n-1$.

2.2.1. Sistemas dinámicos caóticos

El caos es un fenómeno que no es fácil de clasificar o identificar, por esta razón no existe una definición como tal para el caos, sin embargo, varios autores han hecho un esfuerzo para dar una definición de caos determinista en las trayectorias de un sistema dinámico siendo la de Robert Devaney [57] la más aceptada:

Definición 10. *Sea $(X;d)$ un espacio métrico, y sea un mapa $f : X \to X$ una función continua, se dice que el sistema es caótico en X si satisface las siguientes condiciones:*

1. *f tiene sensibilidad a las condiciones iniciales. Esto es, existe un $\varepsilon > 0$ tal que, para cualquier $x \in X$ y $\delta > 0$, existe algún $y \in X$ donde la distancia $d(x;y) < \delta$ y $m \in N = \{1,2,3,\ldots\}$ tal que la distancia $d(f^m(x); f^m(y)) > \varepsilon$.*

2. *f es topológicamente transitiva. Esto es, para cualquier par de conjuntos abiertos $U,V \subset X$, existe cierto $m \in N$ tal que $f^m(U) \bigcap V \neq \emptyset$.*

3. *f tiene una* **"distribución"** *densa de órbitas periódicas. Esto es, supongamos que Y contiene todas las órbitas periódicas de f, entonces para cualquier punto $x \in X$, existe un punto y en el subconjunto Y arbitrariamente cercano a x.*

El concepto de vecindad de un punto $x \in X$ es importante para demostrar la segunda condición de la definición de caos de Devaney y se muestra a continuación:

Definición 11. *La vecindad de un punto $x \in X$ es un conjunto $N_\delta(x)$ que contiene todos los puntos $y \in X$ tal que la distancia $d(x,y) < \delta$. El número δ es conocido como el radio de $N_\delta(x)$.*

En 1992 Banks y colaboradores [58] demostraron que la condición de sensibilidad a las condiciones iniciales es redundante, es decir que la condición de transitividad y las órbitas periódicas densas, juntas implican sensibilidad a las condiciones iniciales. Por otro lado en 1994 Vellekoop y Berglund [59] mostraron que para mapeos continuos, la transitividad implica que el conjunto de órbitas periódicas es denso, con lo cual transitividad implica caos.

Existe otra herramienta que indica si un sistema dinámico es caótico, es conocida como el exponente de Lyapunov y se basa en la condición de sensibilidad a las condiciones iniciales [60]. Básicamente mide la tasa de divergencia exponencial entre trayectorias cercanas, donde d_n es la distancia entre dos órbitas al tiempo n, separadas inicialmente por una distancia d_0

$$\frac{d_n}{d_0} = e^{\lambda n}, \lambda = \frac{1}{n} ln \left| \frac{d_n}{d_0} \right| \tag{2.5}$$

Para que el resultado sea significativo, la divergencia exponencial se debe medir en la vecindad de la trayectoria de referencia; por lo tanto los exponentes de Lyapunov se calculan a cada iteración y se toma la media al cabo de un número N de iteraciones suficientemente grande,

$$\lambda = \lim_{N \to \infty} \frac{1}{N} \sum_{n=0}^{N-1} ln \left| \left(\frac{d_z}{d_0} \right)_n \right| \tag{2.6}$$

donde d_0 y d_z son las distancias inicial y final en la iteración n, λ es el exponente de Lyapunov.

2.2.2. Sistemas dinámicos discretos

Este trabajo de tesis se basa principalmente en sistemas dinámicos discretos también conocidos como mapeos o mapas por esta razón daremos más detalles de este tipo de sistemas. Como se mencionó los sistemas dinámicos de tiempo discreto se describen por medio de ecuaciones en diferencias

$$x_{n+1} = f(x_n), \tag{2.7}$$

este tipo de expresión se les conoce como relación de recurrencia o función iterativa donde x_{n+1} es calculada (iterada) a partir de x_n y $n = \{0,1,2,3,\dots\}$, si se comienza con un valor inicial x_0 entonces sus iteraciones nos describen la órbita:

$$\{x_i : i = 0 \to \infty\} = \{x_0, x_1, x_2, \dots, x_n, x_{n+1}, \dots\}. \tag{2.8}$$

De forma análoga podemos escribir la misma órbita en la siguiente notación:

$$x_1 = f(x_0), x_2 = f(x_1) = f(f(x_0)), \dots, x_n = f^n(x_0). \tag{2.9}$$

Una forma de representar las órbitas de un sistema discreto es por su método gráfico conocido como "cobwebbing", esta técnica consiste en superponer la gráfica $y = x$ sobre la gráfica del mapeo. Comenzando en algún valor inicial x_0 se dibuja una línea vertical hasta la gráfica del mapeo $f(x_n)$ y de este punto se dibuja una línea horizontal a la línea de la gráfica $y = x$. Con esto se tiene el resultado de la primera iteración, este proceso se repite de la misma forma cuantas veces sea necesario, más adelante mostraremos un ejemplo de gráfica cobwebbing.

En secciones anteriores se mostró que la órbita más sencilla es la que se mantiene en equilibrio, si especificamos esta definición para sistemas discretos tenemos que un punto fijo se define de la siguiente forma:

Definición 12. *Un punto fijo, o punto de periodo uno, es un punto en el cual* $x_{n+1} = f(x_n) = x_n$ *para todo n.*

Gráficamente los puntos fijos pueden encontrarse en la intersección de la función $f(x_n)$ con la diagonal $x_{n+1} = x_n$.

Definición 13. *Un punto periódico de periodo N es un punto en el cual* $x_{n+N} = f^N(x_n) = x_n$ *para todo n.*

Es importante notar que la presencia de algún punto fijo estable implica la obtención de órbitas periódicas de periodo 1, mientras que al tener puntos fijos inestables se tiene la posibilidad de tener orbitas de periodo N.

Para encontrar puntos periódicos de periodo dos en un mapeo, es necesario encontrar los puntos donde se intersectan $f^2(x)$ con la diagonal, de forma similar para encontrar puntos fijos de periodo tres, se deben de encontrar los puntos de intersección de $f^3(x)$ con la diagonal. El trabajo de Li-Yorke [18] muestra que si un mapeo contiene puntos periódicos de periodo tres, entonces es posible encontrar puntos periódicos de todos los periodos.

Teorema 1. *[27](Sección 12.3, página 277) El mapeo $f(x)$ tiene un punto fijo en x^*. El punto fijo es estable si:*

$$\left|\frac{d}{dx}f(x^*)\right| < 1, \tag{2.10}$$

El punto fijo es inestable si:

$$\left|\frac{d}{dx}f(x^*)\right| > 1. \tag{2.11}$$

Es importante notar que la presencia de algún punto fijo estable implica la obtención de órbitas periódicas de periodo 1, mientras que al tener puntos fijos inestables se tiene la posibilidad de tener órbitas de periodo N.

En la sección anterior se mostró de forma general como encontrar el exponente de Lyapunov, sin embargo cuando se conoce la expresión del sistema caótico en este caso sistemas discretos $x_{n+1} = f(x_n)$ se tiene lo siguiente:

$$\lambda = f(x_n + d_0) - f(x_n) = \left(\frac{df}{dx}\right)_{x_n} d_0, \tag{2.12}$$

de esta forma obtenemos el exponente de Lyapunov para sistemas discretos como [27]:

$$\lambda = \lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} ln\left|f'(x_n)\right| \tag{2.13}$$

donde f' representa la derivada de la función respecto a x, x_n son iteraciones sucesivas, $x_0, x_1, x_2, \ldots, x_n$. Por lo tanto el exponente de Lyapunov se calcula con base a un conjunto de puntos.

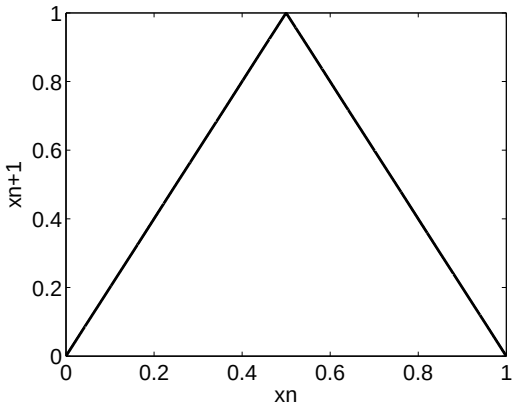


Figura 2.8: Gráfica de la función casa de campaña.

Teorema 2. *[27](Sección 12.3, página 282) Si el exponente de Lyapunov es positivo* $\lambda > 0$*, entonces el sistema es caótico; si el exponente de Lyapunov es negativo* $\lambda < 0$*, entonces la órbita es periódica y cuando el exponente de Lyapunov es cero* $\lambda = 0$*, ocurre una bifurcación.*

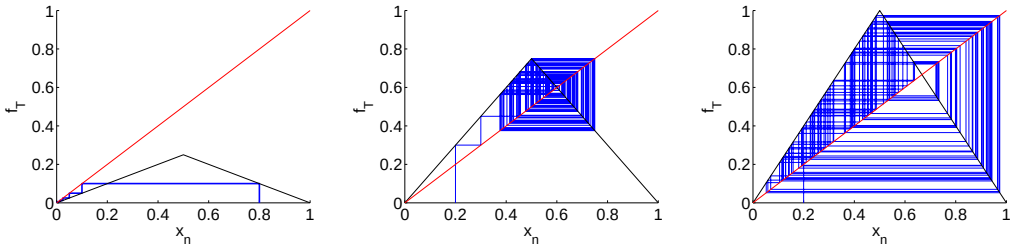
Otra herramienta que se usa para el estudio de sistemas dinámicos son los diagramas de bifurcación, los cuales son una representación gráfica del comportamiento de las órbitas en función de un parámetro. En estos diagramas es fácil ver el periodo de las órbitas y la propiedad que se conoce como cascadas de periodo dos. En la cual puntos de periodo uno se convierten en puntos de periodo dos, posteriormente en puntos de periodo cuatro y así sucesivamente hasta llegar a caos, que es el comportamiento útil para generar secuencias pseudo-aleatorias., más adelante se mostrarán este tipo de diagramas.

2.2.2.1. Sistemas discretos uni-modales

El mapeo casa de campaña (Tent map) y el mapeo Logístico son los sistemas unimodales más famosos y más estudiados, se puede encontrar una amplia variedad de referencias en la literatura en donde se exponen sus propiedades [26, 27, 28].

El mapeo casa de campaña (f_T) se construye a partir de dos líneas rectas, por lo que es una función lineal por partes (ver figura 2.8) se define como se muestra a continuación:

$$f_{T,}(x,\mu) = \begin{cases} \mu x, & \text{para } x < 1/2, \\ \mu(1-x), & \text{para } x \geq 1/2, \end{cases} \tag{2.14}$$



(a) (b) (c)

Figura 2.9: Diagrama cobweb para diferentes valores de μ: a)0.5, b)1.5, c)2.

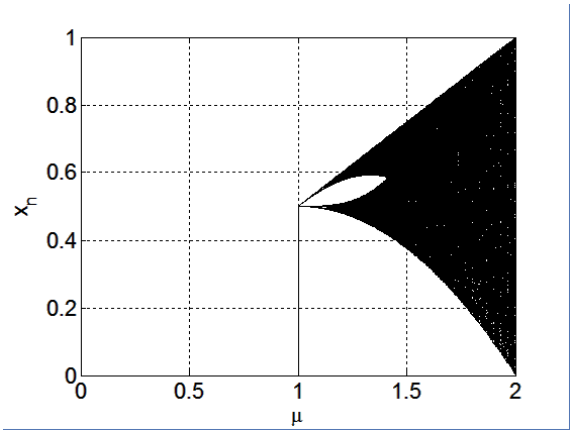


Figura 2.10: Diagrama de bifurcación del mapeo casa de campaña.

Aunque la forma del mapeo casa de campaña es simple e incluye ecuaciones lineales, para ciertos valores de μ el sistema puede mostrar comportamiento complejo e incluso comportamiento caótico. Para poder observar las órbitas del sistema de forma gráfica podemos usar los diagramas "cobweb" como se muestra en la figura 2.9.

Dependiendo del valor del parámetro μ se pueden tener diferentes puntos fijos, cuando $\mu < 1$ se tiene un único punto fijo localizado en 0, cuando $\mu = 1$ una de las rectas trazadas por la función f_T esta sobrepuesta con la recta $y = x$ por lo que se tiene un número infinito de puntos fijos localizados en $0 < x_n < 0.5$. Por último cuando el valor de $\mu > 1$ se tienen dos puntos fijos uno localizado en 0 y el otro en $\frac{\mu}{1+\mu}$.

Por medio del diagrama de bifurcación podemos observar las órbitas que se generan para cada uno de los posibles valores de μ con una condición inicial fija x_0, en la figura 2.10 se muestra el diagrama de bifurcación del mapeo casa de campaña $f_T(x_0, \mu)$.

Por último, en la figura 2.11 se muestra el exponente de Lyapunov para cada uno de

los valores de μ, en la figura podemos observar que para valores de $\mu > 1$, el sistema es caótico.

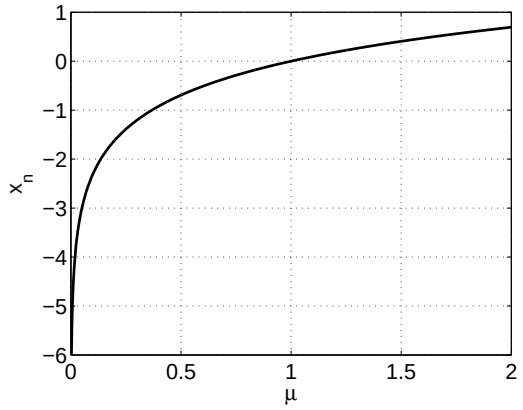


Figura 2.11: Exponente de Lyapunov del mapeo casa de campaña.

Por otro lado, el mapeo logístico (f_L) fue presentado como un modelo de crecimiento de población de especies por Robert May [19], y está definido por la siguiente ecuación:

$$f_L(x,\alpha) = \alpha x(1-x) \tag{2.15}$$

donde $\alpha \in I_p \subset \Re$ es el parámetro de bifurcación. Generalmente, el parámetro de bifurcación α ha sido estudiado en el intervalo $I_+ = [0,4]$. Sin embargo, matemáticamente no se tiene ninguna restricción para tomar valores negativos, por lo que el mapeo logístico también ha sido estudiado para valores negativos en el intervalo $I_- = [-2,0)$. En [29] los autores observaron la dinámica del sistema en estos dos intervalos y encontraron información útil para aplicaciones de acciones, por otro lado estos intervalos fueron útiles para construir un generador de números pseudo-aleatorios [61] el cual se mostrará a detalle en el siguiente capítulo.

Cuando tomamos valores de α entre estos dos intervalos I_+, I_- la órbita no escapa hacia el infinito para cualquier condición inicial. En la figura 2.12 se muestra el mapeo logístico para diferentes valores de $\alpha \in I_p = I_+ \cup I_-$, $\alpha = -2$ línea formada por triángulos, $\alpha = -1.3333$ linea formada por diamantes, $\alpha = -0.6666$ linea formada por cuadrados, $\alpha = 1.3333$ linea formada por asteriscos, $\alpha = 2.6666$ linea formada por círculos, $\alpha = 4$ linea formada por cruces, se puede observar que el sistema presenta uno o dos puntos fijos dependiendo del valor de α los cuales están ubicados en 0 y en $\frac{\alpha-1}{\alpha}$ para $\alpha \neq 0$.

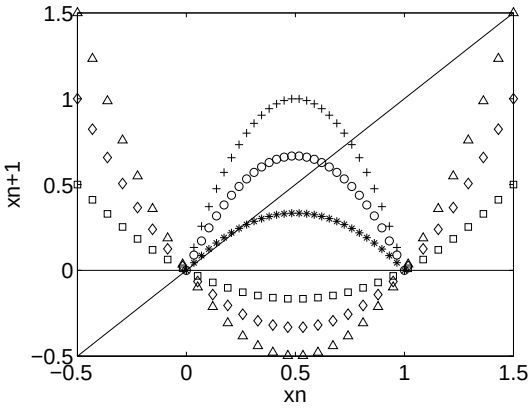


Figura 2.12: Mapeo logístico con diferentes valores de α.

La estabilidad de los puntos fijos puede ser estable o inestable. En la figura 2.13 se muestra la estabilidad de los puntos fijos en función del parámetro de bifurcación α, donde una cruz denota un punto fijo estable o atractivo, mientras que un círculo denota que el punto fijo es inestable o repulsivo.

El punto fijo localizado en 0 es repulsivo para $\alpha \in [-2,-1] \cup [1,4]$ y es atractivo para $\alpha \in (-1,1)$. El segundo punto fijo está localizado en $\frac{\alpha-1}{\alpha}$, es atractivo para $\alpha \in [1,3]$ mientras que es repulsivo para $\alpha \in [3,4]$. Además existe otro punto fijo el cual es repulsivo, está localizado en 1.5 y solo existe cuando $\alpha = 2$.

Un punto fijo atractivo no permite oscilaciones debido a que todas las órbitas convergen al punto, mientras que para un punto fijo repulsivo se presentan órbitas periódicas e incluso órbitas caóticas. La figura 2.14 muestra el diagrama de bifurcación del mapeo logístico $f_L(x_0,\alpha)$ con $\alpha \in [-2,4]$. Es importante remarcar que $f_L : [0,1] \to [0,1]$ para $\alpha \in [0,4]$ mientras que $f_L : [-0.5,1.5] \to [-0.5,1.5]$ para $\alpha \in [-2,0]$. En el diagrama de bifurcación se puede apreciar la cascada de periodo-doble por medio de las bifurcaciones, las órbitas de periodo 1 se convierten en periodo 2, posteriormente en periodo 4, etc. Estas bifurcaciones son las que llevan a un sistema del orden al caos.

El exponente de Lyapunov del mapeo logístico se muestra en la figura 2.15. Se puede observar que la gráfica es simétrica respecto a $\alpha = 1$, por lo que la dinámica del mapeo logístico para los parámetros $\alpha \in [1,4]$ es semejante a la dinámica dada por los parametros $\alpha \in [-2,1]$. Esta simetría se da a pesar de que los puntos fijos son diferentes.

Cuando $\alpha \in (-1,1)$ el sistema solo tiene un punto fijo atractivo localizado en cero y $\lambda < 0$, por lo que cada órbita que se genere converge al punto fijo. Para $\alpha \in [1,3]$ el

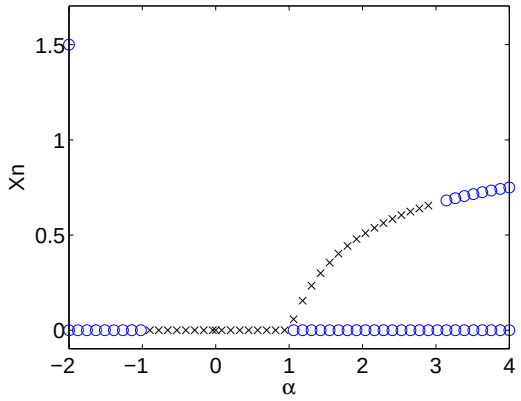


Figura 2.13: Estabilidad de los puntos fijos del mapeo logístico. Cruz ($\times$) y círculo ($\circ$) denotan puntos fijos estables e inestables, respectivamente.

sistema tiene dos puntos fijos: uno atractivo y otro repulsivo, pero $\lambda < 0$ por lo que las órbitas convergen al punto fijo. Para $\alpha = 3$ el sistema presenta una bifurcación y el valor del exponente es $\lambda = 0$. Para $\alpha \in (3,4]$ el sistema tiene dos puntos fijos y ambos son repulsivos, el valor de $\lambda < 0$ cuando las órbitas oscilan de forma periódica y $\lambda > 0$ cuando la órbita oscila de forma caótica. Por último cuando $\alpha = -2$ el sistema tiene dos puntos fijos repulsivos y $\lambda > 0$ por lo que las órbitas oscilan de forma caótica.

2.2.2.2. Sistemas discretos bi-modales

Los mapeos casa de campaña y logístico presentan un punto crítico situado en $c_0 = 0.5$, incrementa para $x \in [0, 0.5)$ y decrementa para $x \in [0.5, 1]$. En esta sección presentaremos un mapeo que posee dos puntos máximos por esta razón es un mapeo bi-modal y es construido a partir de mapeos uni-modales [62]. Para esto se consideran los mapeos casa de campaña y logístico con sus parámetros de bifurcación máximos $\alpha = 4, \mu = 2$, respectivamente. El mapeo $f_D(x,\beta)$ se define como la diferencia o la resta entre estos dos mapeos multiplicados por un nuevo parámetro de bifurcación $\beta \in [0,4]$ de tal forma que se tiene:

$$f_D(x,\beta) = \beta(f_L(x,4) - f_T(x,2));$$

$$f_D(x,\beta) = \begin{cases} 2\beta x(1-2x), & \text{para } x < \frac{1}{2}; \\ 2\beta(x-1)(1-2x), & \text{para } x \geq \frac{1}{2}. \end{cases} \tag{2.16}$$

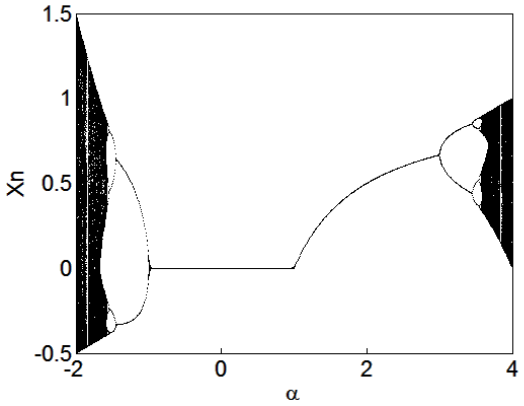


Figura 2.14: Diagrama de bifurcación del mapeo logístico.

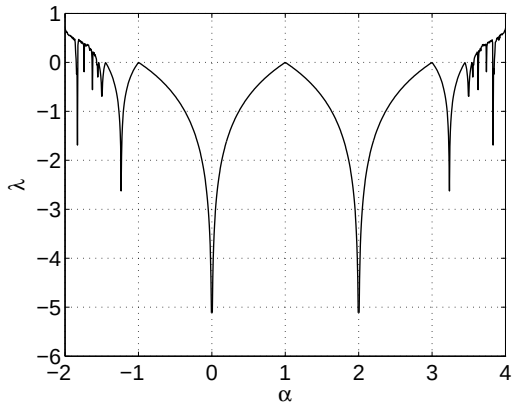


Figura 2.15: Exponente de Lyapunov del mapeo logístico.

El parámetro β amplifica la diferencia entre el mapeo logístico y el mapeo casa de campaña, para $\beta = 4$ el mapeo es bi-modal $f_D(x,\beta) : [0,1] \to [0,1]$. En la figura 2.16 se muestra el mapeo para diferentes valores de β: 1.333 línea formada por círculos, 2.666 línea formada por puntos, 4 línea formada por triángulos, también se puede observar que el sistema presenta dos puntos críticos $c_0 = 0.25$ y $c_1 = 0.75$ localizados en los intervalos $I_0 = [0, 0.5)$ y $I_1 = [0.5, 1]$, respectivamente. El mapeo bi-modal siempre tiene un punto fijo en 0 pero puede presentar otros puntos fijos dependiendo del valor de β y estos se localizan en $\frac{2\beta-1}{4\beta}$, $\frac{6\beta-1-\sqrt{4\beta^2-12\beta+1}}{8\beta}$ y $\frac{6\beta-1+\sqrt{4\beta^2-12\beta+1}}{8\beta}$. La estabilidad de los puntos fijos se muestra en la figura 2.17, donde una cruz denota un punto fijo atractivo, mientras que un círculo denota que el punto fijo es repulsivo.

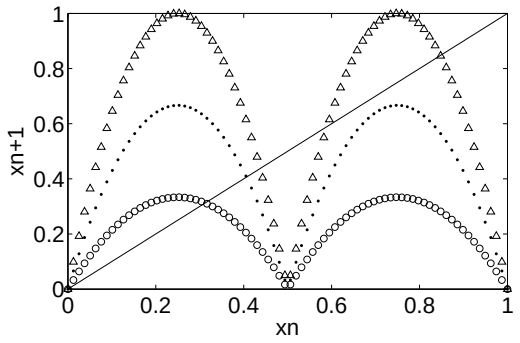


Figura 2.16: Mapeo bi-modal con diferentes valores de β.

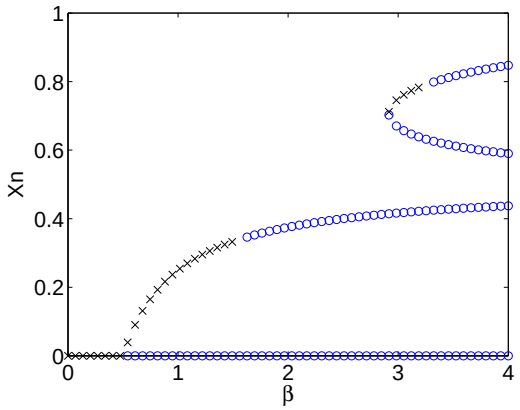


Figura 2.17: Estabilidad de los puntos fijos del mapeo bi-modal. Cruz y círculo denotan puntos fijos estables e inestables, respectivamente.

Una propiedad importante de este mapeo es que puede comportarse como uni-modal o bi-modal dependiendo del valor del parámetro de bifurcación. Si $\beta = 2$, para cualquier condición inicial $x_0 \in [0,1]$, $f_D(x,\beta)$ después de la primera iteración se comporta como un mapeo unimodal $f_D(x,\beta) : [0,0.5] \to [0,0.5]$. Por otro lado si $\beta = 4$, $f_D(x,\beta) : [0,1] \to [0,1]$ se comporta como un mapeo bi-modal.

En la figura 2.18 se muestra el diagrama de bifurcación del mapeo bi-modal $f_D(x_0,\beta)$ el cual se ecuentra en $[0,1] \times [0,4]$. Se pueden observar bifurcaciones que llevan a las secuencias a aumentar de periodo aproximadamente en $\beta = 1.5$ y $\beta = 3.2312$. Para valores de $\beta \in [0,2]$ el sistema se asemeja al mapeo logístico pero oscila en el intervalo de $[0,0.5]$ y para $\beta \in [2,4]$ se comporta como mapeo bi-modal que oscila en el intervalo $[0,1]$.

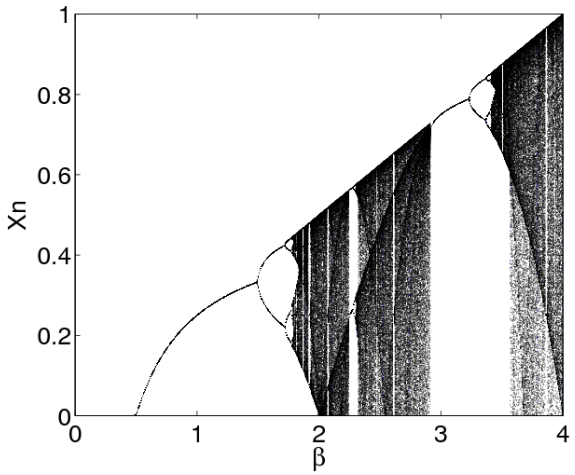


Figura 2.18: Diagrama de bifurcación del mapeo bi-modal.

Teorema 3. *[62] El mapeo bi-modal* $f_D(x,\beta)$ *es caótico en el sentido de Devaney en* $[0,1]$ *para* $\beta = 4$.

Demostración. De acuerdo a la definición 10, se necesitan probar tres condiciones: (1) sensibilidad a las condiciones iniciales, (2) transitividad y (3) la distribución densa de órbitas periódicas.

Comenzaremos por demostrar la última propiedad. Para esto necesitamos probar que existe el subconjunto Y en el intervalo $I = [0,1]$ el cual está formado por las órbitas periódicas y Y es denso en I. El intervalo puede ser dividido en $J_0^1 = [0, c_0^1], J_1^1 = [c_0^1, \eta_0^0 = 0.5], J_2^1 = [\eta_0^0 = 0.5, c_1^1]$ y $J_3^1 = [c_1^1, 1]$ ver figura 2.19, cada intervalo contiene un punto fijo del mapeo bi-modal $f_D, \Delta^1 = \{p_0^1 = 0, p_1^1 = 0.4375, p_2^1 = 0.5899, p_3^1 = 0.8476\}$, respectivamente. Estos puntos fijos en el intervalo cerrado I pertenecen a Y como órbitas periódicas de periodo uno, donde $c_0^1 = 0.25$ y $c_1^1 = 0.75$ son los puntos críticos. Nótese que $f_D : J_i^1 \to [0,1], i = 0,\ldots,3$, entonces cada subintervalo representa al mapeo bi-modal para f_D^2. Podemos ver que $f_D(0) = f_D(0.5) = f_D(1) = 0$ y $f_D(c_0^1 = 0.25) = f_D(c_1^1 = 0.75) = 1$. La observación anterior nos permite inferir que para todo $x \in I$ y si $f_D^k(x) = 0.5$ entonces $f_D^{k+1}(x) = 0$.

Los puntos fijos se encuentran en la intersección entre f_D y la función identidad $f_I(x) = x$. Si consideramos la intersección entre la segunda iteración f_D^2 y f_I encontraremos que estas funciones se intersectan en 16 puntos, que representan a el conjunto de puntos fijos Δ^1 y el conjunto de puntos periódicos de periodo dos Δ^2. Ahora el in-

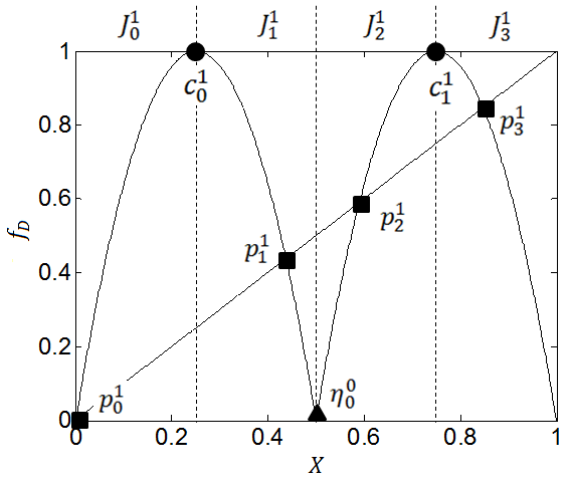


Figura 2.19: Mapeo bi-modal y los subintervalos $J_i^1, i = 0,1,2,3$. Círculos denotan puntos críticos, cuadrados denotan puntos fijos y triángulos denotan los puntos η.

tervalo I está formado por 16 subintervalos $J_0^2 = [0, c_0^2], J_1^2 = [c_0^2, \eta_0^1], J_2^2 = [\eta_0^1, c_1^2], J_3^2 = [c_1^2, c_0^1], J_4^2 = [c_0^1, c_2^2], J_5^2 = [c_2^2, \eta_1^1], J_6^2 = [\eta_1^1, c_3^2], J_7^2[c_3^2, 0.5], J_8^2 = [0.5, c_4^2], J_9^2 = [c_4^2, \eta_2^1], J_{10}^2 = [\eta_2^1, c_5^2], J_{11}^2 = [c_5^2, c_1^1], J_{12}^2 = [c_1^1, c_6^2], J_{13}^2 = [c_6^2, \eta_3^1], J_{14}^2 = [\eta_3^1, c_7^2]$ y $J_{15}^2 = [c_7^2, 1]$. La figura 2.20 muestra los subintervalos $J_i^2, i = 0,1,2,\ldots,15$, los puntos fijos están marcados con cuadrados y los puntos periódicos de periodo dos están marcados con asteriscos. $\Delta^2 = p_0^2, p_1^2, p_2^2, p_3^2, \ldots, p_{11}^2$. El conjunto $\{c_0^2, c_1^2, c_2^2, c_3^2, c_4^2, c_5^2, c_6^2, c_7^2\}$ contiene los puntos críticos de f_D^2 y $\eta_i^1 = f_D(x) = 0.5, i = 0,\ldots,3$. Los puntos periódicos de periodo uno y dos pertenecen a $Y \supset \Delta^1 \cup \Delta^2$. En general, las intersecciones entre f_D^n y f_I muestran los puntos periódicos de periodo n y pueden ser puntos periódicos de menor periodo. I está comprendido por subintervalos $J_i^n, i = 0,\ldots,4^n - 1$ y los puntos finales de cada intervalo están dados por los puntos críticos de $f_D^k, \eta^{k-1} = f_D^{k-1}(x) = 0.5, k = 1,\ldots,n$. Cada subintervalo J_i^n contiene al menos un punto periódico y además al buscar puntos periódicos de periodo n podemos dividir I en $4^n - 1$ subintervalos, por lo tanto cuando se buscan puntos de periodo muy grande $n \to \infty$ los subintervalos $|J_i^n| \to 0$. Por lo que para cualquier $x \in I$, existe un punto y en el subconjunto Y arbitrariamente cercano a x, esto prueba que los puntos periódicos son densos en $[0,1]$.

Para demostrar que f_D es topológicamente transitiva, consideraremos un par de conjuntos abiertos $N_\delta(y_1), N_\delta(y_2) \subset I$, para cualquier $y_1, y_2 \in I$, necesitamos mostrar que existe un cierto $m \in N = \{1,2,3,\ldots\}$ tal que $f_D^m(N_\delta(y_1) \cap N_\delta(y_2)) \neq \emptyset$, dicho de otra forma necesitamos demostrar que al menos una órbita con una condición inicial $x_0 \in N_\delta(y_1)$

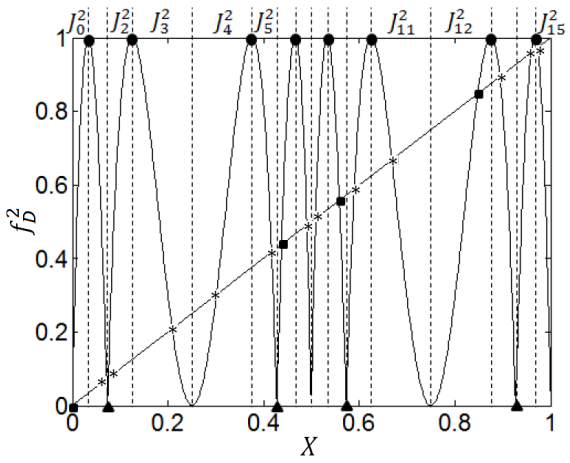


Figura 2.20: Mapeo bi-modal y los subintervalos $J_i^1, i = 0,1,2,\ldots,15$. Circulos denotan puntos criticos, cuadrados denotan puntos fijos y triangulos denotan η.

evoluciona a $N_\delta(y_2) \ni f_D^m(x_0)$. Para esto consideremos los dos conjuntos abiertos $N_\delta(y_1)$ y $N_\delta(y_2)$ localizados arbitrariamente en I como se muestra en la figura 2.21. En párrafos anteriores se discutió que cada subintervalo J_i^k tiende a cero cuando k tiende a infinito, además sabemos que cada subintervalo J_i^k es mapeado sobre el intervalo I, $f_D^k : J_i^k \to I$. Consideremos el subintervalo $J_i^m \subset N_\delta(y_1)$ como se muestra en las figuras 2.22 y 2.23. En consecuencia $f(J_i^m) = I \supset N_\delta(y_2)$ entonces $f_D^m(x_0) \in N_\delta(y_2)$, para cualquier $x_0 \in N_\delta(y_1)$, esto prueba que f_D es topológicamente transitiva.

Finalmente, falta por demostrar la sensibilidad a las condiciones iniciales, para esto comenzaremos por definir $\varepsilon = |I|/2$, donde $|I| = 1$, tal que para cualquier $x_{01} \in I$ y cualquier $\delta > 0$ existe un $x_{02} \in N_\delta(x_{01})$ tal que la distancia entre $|f_D^m(x_{01}) - f_D^m(x_{02})| \geq \varepsilon$.

Si consideramos el subintervalo J_i^{m-1} tal que $J_i^{m-1} \subset N_\delta(x_{01})$ entonces existe un $x_{02} \in J_i^{m-1}$ tal que $|f_D^m(x_{01}) - f_D^m(x_{02})| \geq 1/2$. Por lo tanto tenemos sensibilidad a las condiciones iniciales, es importante remarcar que la definición de sensibilidad a las condiciones iniciales no requiere que la órbita de x_{02} se mantenga alejada de x_{01} para todas las iteraciones, solo necesitamos que un punto en la órbita este alejado de x_{01} en la iteración correspondiente. □

Por otro lado en la figura 2.24 se muestra el exponente de Lyapunov denotado por λ del mapeo bi-modal. Para $\beta \in [0, 0.5]$ el sistema tiene solo un punto fijo atractivo y $\lambda < 0$, por lo que la órbita converge al punto fijo. Para $\beta \in [0.5, 1.5)$ el sistema tiene dos puntos fijos: uno atractivo y otro repulsivo además $\lambda < 0$ con esto la órbita converge

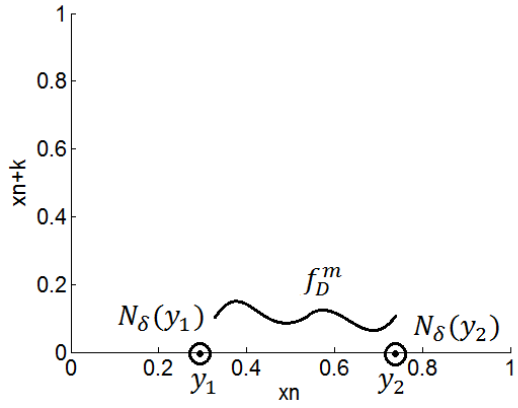


Figura 2.21: Existe una órbita tal que dos puntos se acercan arbitrariamente.

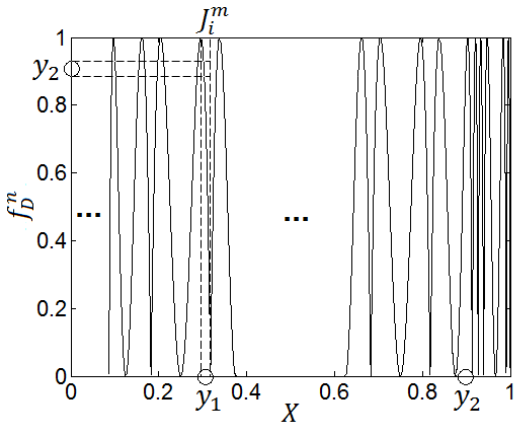


Figura 2.22: Transitividad de una órbita de periodo $n(f_D^n)$ del mapeo bi-modal.

al punto atractivo, sin embargo cuando $\beta = 1.5$ se presenta un bifurcación y el valor de $\lambda = 0$. Para $\beta \in (1.5, 2.915)$ el sistema tiene dos puntos fijos repulsivos, cuando $\lambda < 0$ las órbitas oscilan de forma periódica y cuando $\lambda > 0$ las órbitas oscilan de forma caótica. Para $\beta \in (2.915, 3.235)$ el sistema tiene cuatro puntos fijos, tres de ellos son repulsivos y uno atractivo, por esto la órbita converge al punto fijo atractivo y $\lambda < 0$, además cuando $\beta = 3.235$ se presenta otra bifurcación por lo que $\lambda = 0$. Para $\beta \in (3.235, 4]$ el sistema mantiene cuatro puntos fijos pero ahora todos son repulsivos. La órbita oscila de forma periódica o caótica cuando $\lambda < 0$ y $\lambda > 0$, respectivamente.

Una parte de este trabajo de tesis estuvo enfocada en realizar la implementación experimental del mapeo bi-modal por medio de un circuito analógico, dicha implementación

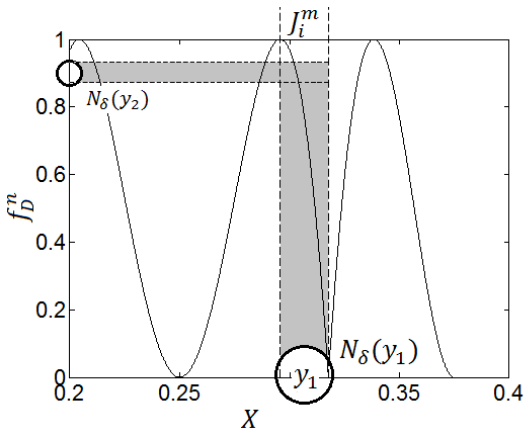


Figura 2.23: Zoom de la figura 2.22, en donde se aprecia la transitividad de una órbita de periodo *n* donde $f(J_i^m) \supseteq I \supset N_\delta(y_1)$.

se encuentra en el apéndice A. Esta realización experimental puede servir para construir un generador de números aleatorios (TRNG), y se caracteriza por el hecho de que su salida no puede ser reproducida debido al ruido presente en toda implementación y es útil para construir llaves para esquemas de cifrado.

2.2.2.3. Sistemas discretos multi-modales

La idea básica del mapeo multi-modal es extender el concepto del mapeo bi-modal y construir un mapeo de cualquier número de modas, a este mapeo también se le conoce como *k*-modal. Dicho de otra forma un mapeo multi-modal contiene k puntos críticos denotados por $c_0, c_1, \ldots, c_{k-1}$ en $I = [a,b] \subset \Re$, crece monótonamente a la izquierda de cada c_i y decrece monótonamente a la derecha de cada c_i, por lo que puede ser descrito como una composición de k mapeos uni-modales $f_1, f_2, \ldots, f_k$, el intervalo I debe de ser particionado en k subintervalos y la familia $\mathcal{F}$ de mapeos está gobernada por un parámetro de bifurcación β. Recordemos que el punto crítico c de un mapeo suave por partes $f(x)$: $I \to I$ es $c \in I$ donde f es diferenciable y $f'(c) = 0$ o cuando $f'(c)$ no existe, sin embargo en un mapeo suave siempre se presentará el caso donde $f'(c) = 0$.

Los mapeos milti-modales fueron presentados en [30], a continuación se presenta una definición formal de los mapeos *k*-modales [62]:

Definición 14. *El mapeo $f : I \to I$ es llamado k-modal, si es continuo en I y tiene k puntos críticos denotados por $c_0, c_1, \ldots, c_{k-1}$ en I. Además existen intervalos $I_i, i =$*

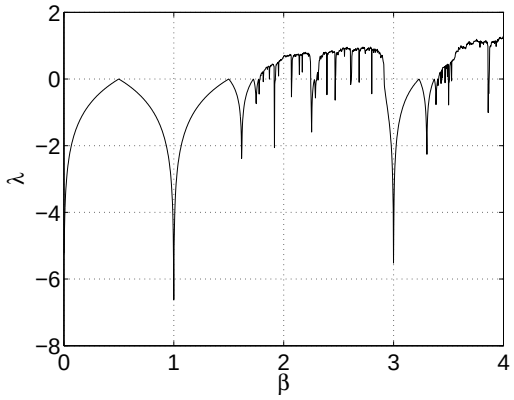


Figura 2.24: Exponente de Lyapunov del mapeo bi-modal.

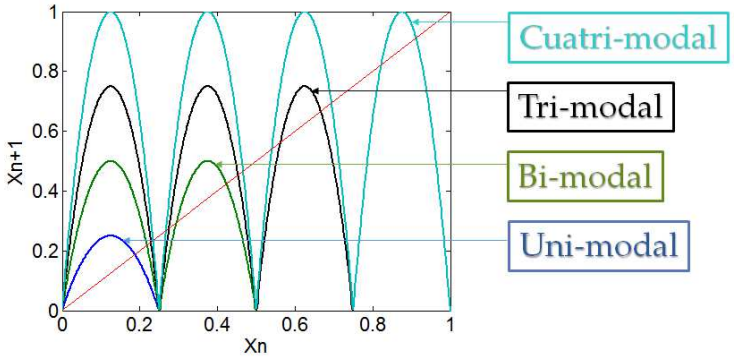


Figura 2.25: Familia $\mathcal{F}$ de mapeos cuatri, tri, bi y uni-modales basados en la forma del mapeo logístico.

$0,\ldots,k-1,\cup_{i=1}^{k} I_{i-1} = I$, *tal que* $\forall\, i = 0,\ldots,k-1$ *se cumple* $c_i \in I_i$ *y* $f(c_i) > f(\beta,x), \forall\, x \in I_i$ *y* $x \neq c_i$, *donde* β *es un parámetro. El caso* $k = 1$ *es el más sencillo y es llamado mapeo uni-modal.*

La definición anterior no restringe a la función f a tener solamente k puntos críticos, sin embargo solo se consideran aquellos que son máximos locales en un subintervalo. Donde el intervalo $I = [a,b]$ dividido en $I_0 = [d_1,d_2), I_1 = [d_2,d_3),\ldots, I_{k-2} = [d_{k-2},d_{k-1}), I_{k-1} = [d_{k-1},d_k]$ por lo tanto el sistema f_β es una función por partes de k mapeos uni-modales.

La familia $\mathcal{F}$ de mapas f_β está definida por la siguiente función:

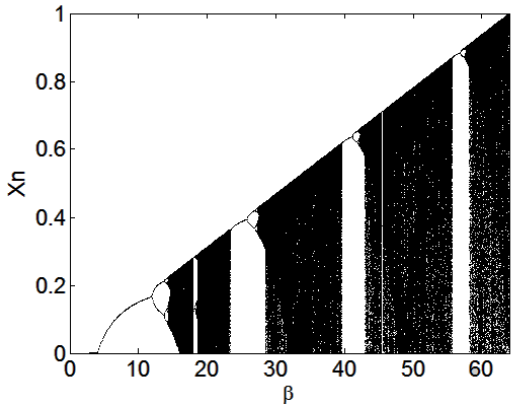


Figura 2.26: Diagrama de bifurcación del mapeo cuatri-modal.

$$f_\beta = \beta(d_{r+1} - x)(x - d_r), x \in I_r \tag{2.17}$$

donde $d_r = r/k, (r = 0, 1, 2, \ldots, k-1)$, k es el número de modas, $\beta = \beta(k, \gamma)$ es el parámetro de bifurcación, $\gamma = 1/k$ es la capacidad de carga. Se puede observar que $I = \cup_{r=0}^{k-1}[dr, d_{r+1})$ y $\cap_{r=0}^{k-1}[d_r, d_{r+1}) = \emptyset$. Para obtener el máximo parámetro de bifurcación con k modas existe una relación directa, $\beta_{max} = (4)(k)/\gamma$.

Para construir un mapeo con $k = 4$ tomaremos la ecuación 2.17, se tendrán 4 subintervalos $I_r(r = 0, 1, 2, 3)$ por lo que la función f_β está dada por la siguiente ecuación:

$$f_\beta(x) \quad = \quad \begin{cases} \beta(1/4 - x)x, & \text{para } x \in [0, 1/4); \\ \beta(1/2 - x)(x - 1/4), & \text{para } x \in [1/4, 1/2); \\ \beta(3/4 - x)(x - 1/2), & \text{para } x \in [1/2, 3/4); \\ \beta(1 - x)(x - 3/4), & \text{para } x \in [3/4, 1], \end{cases} \tag{2.18}$$

donde $\beta \in [0, 64]$ este intervalo está determinado por $k = 4, \gamma = 0.25$.

Una propiedad importante de estos sistemas es que dependiendo del valor de β el sistema se puede comportar como un mapeo: uni-modal $f_{16} : [0, 1] \to [0, 0.25]$, bi-modal: $f_{32} : [0, 1] \to [0, 0.5]$, tri-modal: $f_{48} : [0, 1] \to [0, 0.75]$, cuatri-modal: $f_{64} : [0, 1] \to [0, 1]$. Esto se muestra en la figura 2.25, también se puede observar en la figura que los puntos críticos están localizados en $c_0 = 0.125, c_1 = 0.375, c_2 = 0.625, c_3 = 0.875$. El número y la localización de los puntos fijos varían según el parámetro de bifurcación. Para f_{16} se tienen dos

puntos fijos localizados en $\{0, 0.1875\}$. El sistema bi-modal f_{32} tiene cuatro puntos fijos localizados en $\{0, 0.2188, 0.2950, 0.4238\}$. El mapeo tri-modal tiene seis puntos fijos localizados en $\{0, 0.2292, 0.2756, 0.4536, 0.5625, 0.6667\}$. Finalmente los ocho puntos fijos del mapeo cuatri-modal f_{64} están localizados en $\{0, 0.2344, 0.4663, 0.5402, 0.6941, 0.8223, 0.9121\}$.

En la figura 2.26 se muestra el diagrama de bifurcación del mapeo multi-modal en términos del parámetro β. Consideremos la primera ecuación del sistema f_β (2.18) la cual está dada por $\phi_{I_0}(x) = \beta(1/4 - x)x$, cuando el parámetro β varía de 0 a 16, el diagrama de bifurcación del mapeo ϕ_{I_0} se asemeja al diagrama de bifurcación del mapeo logístico cuando se varía el parámetro α de 0 a 4. Cuando $0 < \beta < 4$ la órbita se estabiliza en el punto fijo localizado en cero independientemente de la condición inicial, mientras que para $4 < \beta < 12$ la órbita se estabiliza en el punto fijo que tiene un valor de $(\beta - 1/\beta)$. Cuando $12 < \beta < 4 + 4\sqrt{6}$, el mapeo presenta órbitas periódicas de periodo dos. La cascada de bifurcaciones de periodo doble comienzan en $\beta = 4 + 4\sqrt{6}$ y terminan en comportamiento caótico en $\beta \approx 14.28$. Para valores mayores a $\beta = 16$ la dinámica continua siendo caótica hasta que aparece un nuevo punto fijo en $\beta = 12 + 8\sqrt{2}$ dando lugar a una nueva cascada de bifurcaciones de periodo doble y este mecanismo se repite como se puede observar en la figura 2.26.

Para demostrar que el sistema cuatri-modal f_β es caótico se puede usar la definición de Devaney y se tomaría como base la demostración presentada con el mapeo bi-modal y simplemente se expande para $k = 4$ modas. Por otro lado en la figura 2.27 se muestra el exponente de Lyapunov en la que podemos ver que cuando $\lambda > 0$ el sistema presenta comportamiento caótico.

2.3. Criptografía caótica

Existen varios trabajos en donde se expone la relación que existe entre estas dos áreas [63, 64, 65]. Como se mencionó anteriormente los conceptos de confusión y difusión son esenciales en la criptografía. Por un lado la ergodicidad es una fuente de confusión, esto es debido a que los puntos que forman las trayectorias, viajan por todo el espacio fase de forma no ordenada sin importar la entrada, de tal forma que para cualquier entrada se tiene la misma distribución de salida. Por otro lado la sensibilidad a las condiciones iniciales y a los parámetros de bifurcación generan difusión, ya que un pequeño cambio en la entrada se transforma en un gran cambio en la salida, al realizar una modificación muy pequeña

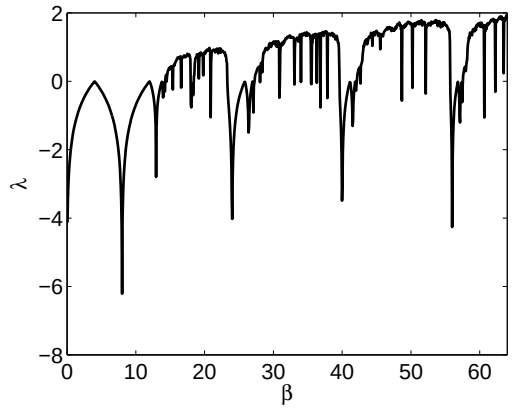


Figura 2.27: Exponente de Lyapunov del mapro cuatri-modal.

en la condición inicial de un sistema caótico se tendrían órbitas muy diferentes mientras que en los criptosistemas un cambio muy pequeño en la llave o en el texto plano llevarían a salidas diferentes.

Los sistemas dinámicos poseen la propiedad de ser deterministas, esto es, la evolución de un sistema a partir de una condición inicial se puede reproducir siempre y cuando se tengan las mismas condiciones iniciales y parámetros, esto se traduce en comportamiento pseudo-aleatorio ya que es posible reproducir exactamente el mismo comportamiento bajo ciertas condiciones. Podemos decir que la similitud de propiedades entre estos sistemas es directa ya que sus propiedades son análogas como se muestra a continuación:

- Sensibilidad respecto a la llave: al invertir un bit de la llave el texto cifrado es completamente diferente.
- Sensibilidad respecto al texto plano: al invertir un bit del texto plano se genera un texto cifrado completamente diferente.
- No debe existir ningún patrón en el texto cifrado el cual pueda relacionar al texto plano.
- Sensibilidad al parámetro de bifurcación: Una pequeña variación en el parámetro es suficiente para generar trayectorias diferentes incluso si comienzan en la misma condición inicial.
- Sensibilidad a las condiciones iniciales: dos trayectorias que comienzan en puntos iniciales muy cercanos, se separan exponencialmente una de otra.

Tabla 2.1: Comparación entre las propiedades caóticas y criptográficas.

Propiedad caótica.	Propiedad criptográfica.	Descripción.
Ergodicidad.	Confusión.	La salida tiene la misma distribución para cualquier entrada.
Sensibilidad a las condiciones iniciales y parámetro de bifurcación.	Difusión.	Un pequeño cambio en la entrada puede causar un gran cambio en la salida.
Dinámica determinista.	Dinámica pseudo-aleatoria.	Un proceso determinista puede causar comportamiento pseudo-aleatorio.
Estructura compleja.	Complejidad del algoritmo.	Un proceso simple puede presentar alta complejidad.

- Ergodicidad: los puntos que forman las trayectorias en el espacio fase están uniformemente distribuidos.

Un resumen de esta comparación se muestran en la tabla 2.1.

Sin embargo aún existe un largo camino por recorrer ya que hay algunas diferencias en el funcionamiento de los criptosistemas clásicos y los criptosistemas basados en caos. En los criptosistemas clásicos la llave está definida por conjunto de bits en un espacio de llaves discreto, mientras que en los sistemas basados en caos la llave está definida por números reales en un espacio de llaves continuo, además en los sistemas clásicos la confusión y la difusión en los sistemas clásicos se logra por medio de varias vueltas de un algoritmo mientras que en los sistemas basados en caos se logra por medio de iteraciones.

Una diferencia importante radica en que el proceso de cifrado en los criptosistemas clásicos está definido sobre conjuntos finitos y en tiempo discreto, mientras que el comportamiento caótico evoluciona sobre números reales y puede desarrollarse en tiempo continuo o discreto, esto ha llevado a que la criptografía caótica se divida en dos ramas las cuales estudiaremos a continuación. En la tabla 2.2 se muestran las principales similitudes y diferencias entre mapeos caóticos y algoritmos criptográficos [66].

Los cifrados que se basan en sistemas caóticos continuos se les denomina sistemas

Tabla 2.2: Similitudes y diferencias entre sistemas caóticos y criptográficos.

Algoritmos criptográficos.	Sistemas caóticos.
Espacio fase: conjunto finito de enteros.	Espacio fase: subconjunto de números reales.
Métodos algebraicos.	Métodos analíticos.
Vueltas.	Iteraciones.
Llave (Booleana)	Parámetros (reales)
Espacio de llaves discreto.	Espacio de llaves continuo.
Realizaciones digitales por medio de aritmética con enteros.	Realizaciones digitales por medio de aritmética con punto flotante.

de comunicación seguros y en general utilizan las técnicas de sincronización de caos desarrollas en los 90's por Pecora y Carroll [25]. En estos sistemas la información se transmite por medio de una señal caótica, mientras que para recuperar la información el transmisor y el receptor deben de sincronizarse, de esta manera un tercero no es capaz de obtener la información.

Por un lado existen diversas formas de transmitir la información entre las cuales se puede mencionar: enmascaramiento caótico [67, 68, 69, 70], conmutación o también conocido como corrimiento de llave (CSK) [71, 72, 73, 74], modulación [75, 76, 77, 78], sistema inverso [79, 80, 81, 82], etc. En la siguiente referencia se trata a fondo el tema de sistemas de comunicación basados en caos [83].

Todos los esquemas anteriores están basados en el fenómeno de sincronización. Existen diferentes modos de sincronización tales como: sincronización completa [25], sincronización generalizada [84, 85, 86], sincronización impulsiva [87, 88, 89], sincronización de fase [90, 91, 92], sincronización proyectiva [93, 94, 95], sincronización con retardo [96, 97, 98], etc. En la siguiente referencia se trata a fondo el tema de sincronización [99].

Los cifrados que se basan en sistemas caóticos discretos se les denominan criptosistemas caóticos o cifrados caóticos digitales, estos no dependen de la sincronización y son diseñados para ser implementados en procesadores digitales, lo cual lleva a un problema de implementación ya que al tener precisión finita su dinámica se ve alterada y no se cumple la definición de Devaney por lo que se dice que presentan caos digital.

A diferencia de los cifrados basados en sistemas continuos, existe una gran variedad de formas y algoritmos para cifrar la información, de forma general podemos tener la

siguiente clasificación:

- Generadores de bits pseudo-aleatorios: se basan en las órbitas que pueden generar los mapeos, por medio de alguna transformación se convierten los valores reales a valores binarios y así generar una secuencia que se le conoce como key-stream. Revisiones de este tema se pueden encontrar en [100, 101].

- Cifrados en flujo: están basados en los generadores de bits, toman como entradas el texto plano y la secuencia de bits (keystream) de tal forma que para cifrar la información se usa la operación XOR y a la salida se tiene el texto cifrado [102, 103, 104, 105].

- Cifrados en bloque: este tipo de cifrados basan su funcionamiento en cajas de sustitución conocidas como S-box, por un lado se pueden definir estas cajas de sustitución por medio de mapeos [106, 107, 108, 109] y por otro lado definir algoritmos que usen dichas cajas y cifren la información por medio de sustitución y permutación [110, 111, 112, 113].

- Cifrados de sustitución y difusión: recientemente se han propuesto varios algoritmos, este tipo se basan en el principio de los cifrados en bloque sin embargo logran el proceso de cifrado en solo una vuelta [114, 115, 116], en las siguientes referencias se realiza un análisis a fondo de este tipo de cifrados [117, 118].

En estos algoritmos la condición inicial y los parámetros de bifurcación son usados como llave del sistema, además la principal ventaja de este tipo de cifrados basados en sistemas discretos radica en que el proceso de cifrado se puede aplicar a cualquier archivo multimedia como son, texto, imágenes, audio, video. Existen revisiones en donde tratan a fondo el tema de cifrados basados en sistemas discretos entre las que se pueden mencionar [119, 120, 121, 122].

2.3.1. Criptoanálisis caótico

El criptoanálisis es una parte integral en el diseño de un algoritmo cifrado, antes de diseñar un nuevo cifrado basado en caos, es esencial tener en cuenta la cantidad de ataques existentes así como las herramientas del criptoanálisis. Como se mostró existe un gran variedad de cifrados caóticos propuestos, debido a esto existe de igual forma una gran variedad de ataques. Cada uno de estos ataques trata de aprovechar las debilidades que

son específicas de cada cifrado en particular, por lo que es complicado diseñar un ataque que pueda ser aplicado a un gran rango de cifrados.

Solo por mencionar algunos trabajos criptoanalíticos, los podemos clasificar de una forma muy general: cifrados basados en sincronización [123, 124, 125, 126, 127], generadores de bits pseudo-aleatorios [128, 129, 130, 131], cifrados por confusión-difusión [132, 133, 134, 135], cifrados por bloques [137, 138, 139, 140].

Esta diversidad de técnicas y algoritmos va en contra de la seguridad de los sistemas basados en caos, ya que en lugar de utilizar estructuras bien analizadas, existe una tendencia general de diseñar nuevas estructuras, de tal forma que se abren caminos para nuevos ataques [136]. Además en varios cifrados propuestos se observa una tendencia al uso de pruebas estadísticas para demostrar la fortaleza del cifrado, si bien esta es una condición necesaria, el hecho de tener buenos resultados estadísticos no significa que se tenga un cifrado que resista a ataques, por lo que es importante realizar varios ataques además de las pruebas estadísticas correspondientes.

Para solucionar estos problemas se han sugerido una serie propiedades básicas que debe de tener cualquier cifrado basado en caos [63, 121, 141, 142, 143]. Las cuales se pueden clasificar en tres: selección del sistema caótico, arquitectura del cifrado e implementación del cifrado.

- Problemas con la selección del sistema caótico.

Definición de una llave que evolucione a comportamiento no caótico: En algunos criptosistemas basados en caos el parámetro de control (o parte del) determina la llave del sistema. Si la relación entre la llave y el parámetro no se establecen cuidadosamente, es posible que el sistema caótico evolucione de forma no caótica, lo que llevará a un bajo rendimiento del criptosistema. Este tipo de problemas se analizan en [144, 145, 146].

Reconstrucción del espacio fase del mapeo: El texto cifrado de algunos criptosistemas puede hacer posible la reconstrucción del espacio fase del mapeo utilizado. Si el espacio fase reconstruido es significativo, entonces es posible identificar los parámetros de bifurcación bajo los cuales evoluciona el sistema. La forma más directa de reconstruir el espacio fase es por medio de la gráfica x_n vs x_{n+1} en donde se puede estimar de forma sencilla los valores del parámetro de bifurcación. Esta técnica de reconstrucción se ha utilizado en [147, 148]. Una solución a este tipo de ataques es mezclar la órbita caótica antes de usarla para cifrar con esto se mezcla el espacio fase del mapeo.

Baja sensibilidad a la llave: El problema más común en los criptosistemas basados en sistemas continuos es la baja sensibilidad a la llave, esta baja sensibilidad es un requerimiento necesario para implementaciones electrónicas con componentes análogos ya que es imposible asegurar una exacta correspondencia en el sistema maestro y el sistema esclavo ya que existe ruido inherente y tolerancia a errores en los valores de los componentes [149, 150].

Erosión de la eficiencia computacional debido a la complejidad estructural del sistema caótico: La complejidad estructural de un sistema caótico es un elemento crítico cuando se evalúa su viabilidad para aplicaciones criptográficas, esta complejidad puede ser minimizada seleccionando sistemas caóticos de tiempo discreto. En un sistema de tiempo discreto se puede construir su espacio fase con una dimensión mientras que para sistemas continuos se requiere de al menos tres dimensiones [152] (entre otras).

- Problemas con la arquitectura de cifrado.

Parte de la llave no debe filtrar el resto de llave: En algunos criptosistemas la llave secreta está formada por varias subllaves. Si el conocimiento de alguna subllave permite conocer el resto de la llave, entonces se puede aplicar un ataque de reconstrucción de llave. En las siguientes referencias se aplica un ataque de este tipo [152, 153, 154].

Estimación de la llave a partir del texto cifrado: Este problema se presenta cuando el texto cifrado está dado por fragmentos de una órbita, versiones muestreadas de la órbita o versiones discretizadas de la órbita. La estimación de parámetros es un problema crítico en los sistemas basados en modulación, ya que con técnicas de sincronización adaptiva se puede obtener una aproximación de los parámetros de control del sistema caótico [145, 155, 156].

Extracción directa del texto plano: Los criptosistemas basados en sistemas caóticos de tiempo continuo, en algunos casos es posible inferir el texto plano a partir de las señales del sistema dinámico sin la necesidad de estimar la llave secreta. Técnicas como filtro de espectro de potencia y reconstrucción de espacio fase pueden ser usadas para este propósito. Aun cuando el espectro de potencia de algunos sistemas caóticos parece ser bueno, picos significativos pueden ser encontrados removiendo las simetrías de los atractores caóticos [157, 158].

Cuando la eficiencia del criptosistema depende del valor de la llave: Algunas arquitecturas de cifrado realizan la transformación del texto plano a texto cifrado por medio

de varias vueltas al algoritmo, en donde en cada vuelta el mapeo caótico es iterado *n* veces. El proceso de cifrado y descifrado deben realizarse en un tiempo constante y que no depende de la llave, por esta razón el número de vueltas no debe depender de la llave de otra forma podría realizarse un ataque de tiempo [159, 160, 161].

Derivación defectuosa de los parámetros del sistema caótico de la llave: En algunos criptosistemas la llave es usada para establecer los valores de los parámetros necesarios para iterar el sistema caótico. Si este proceso implica una reducción del espacio de llaves, entonces un ataque de fuerza bruta podría demandar menos recursos ya que se aplica solo a un subconjunto del espacio de llaves. Una posible solución es usar los parámetros de control y las condiciones iniciales como llave secreta o parte de la llave [162, 163].

Procedimiento de cifrado depende solamente del mapeo: En algunos esquemas de cifrado la transformación del texto plano a texto cifrado se lleva a cabo por un mapeo y una llave, y así generar alguna secuencia, en estas situaciones es posible estimar la llave o una función que sea equivalente al procesado de cifrado. Debido a que las secuencias generadas permanecen sin cambios a menos que se modifique la llave, es posible reconstruir las secuencias por un ataque de texto-escogido [164, 165].

Baja sensibilidad a los cambios en el texto plano: Este problema es especialmente relevante cuando se consideran imágenes. El esquema de cifrado debe garantizar que si dos imágenes que sean diferentes en solo un pixel, sean totalmente diferentes en las imágenes cifradas. Por lo que si se tienen dos imágenes muy similares y de igual forma las imágenes cifradas son similares no se cumple el concepto de difusión.

- Problemas de implementación.

Degradación de la eficiencia por el uso de sistemas caóticos de tiempo continuo: Al utilizar criptosistemas basados en sistemas caóticos de tiempo continuo es necesario utilizar métodos numéricos para obtener las órbitas, estos métodos numéricos incrementan el tiempo y por lo tanto reducen la eficiencia del cifrado [166].

Esquemas de cifrado no invertibles: El proceso de iteración de sistemas caóticos implica la evolución del sistema con números reales, debido a que las implementaciones de los sistemas caóticos se realiza con precisión finita, es posible que las órbitas generadas para el proceso de cifrado no puedan ser regeneradas exactamente para el proceso de descifrado y como consecuencia no se puede recuperar el texto plano aun cuando se conoce la llave del sistema [167].

Degradación dinámica: La implementación de sistemas caóticos en precisión finita en algunos casos lleva a un comportamiento y propiedades dinámicas diferentes a las teóricas, si esto no es tomado en cuenta puede implicar una reducción en el desempeño e incluso comprometer la seguridad del criptosistema. En las siguientes referencias se muestran las consecuencias de esta degradación [168, 169].

Capítulo 3

Generadores pseudo-aleatorios caóticos

Como se mencionó anteriormente, en este trabajo de tesis nos enfocamos en el estudio y desarrollo de generadores pseudo-aleatorios y su aplicación en cifrados en flujo, en el capítulo anterior se definieron diferentes tipos de generadores de secuencias. En este capítulo se muestra el desarrollo de dos nuevos PRNG, para esto se tomaron en cuenta herramientas como el exponente de Lyapunov y la reconstrucción del espacio fase, además para analizar y caracterizar las secuencias generadas estas son evaluadas por medio de pruebas estadísticas. Existe una gran variedad de bancos de pruebas entre los que podemos mencionar la suite desarrollada por Beker y Piper [170], las propuestas por Gustafson [171], las pruebas desarrolladas por Marsaglia [172], sin embargo las pruebas más usadas por la comunidad científica y que forman parte de un estándar, fueron propuestas y desarrollas por el NIST [173], esta suite contiene una variedad de pruebas estadísticas independientes las cuales se exponen a detalle en el apéndice B de tal forma que en este capítulo solo se presentan los resultados de dichas pruebas.

3.1. Generador pseudo-aleatorio basado en series de tiempo con retardo

En esta sección se propone un algoritmo para la construcción de un PRNG que sea seguro para su uso en criptografía, siguiendo el trabajo de Li [41] en el cual se menciona que un generador debe estar basado en la combinación de al menos dos series de tiempo caóticas. Este algoritmo se basa en dos series de tiempo del mapeo logístico $M1, M2$, para esto se requiere de dos parámetros de bifurcación α_1, α_2, dos condiciones iniciales x_{01}, x_{02}

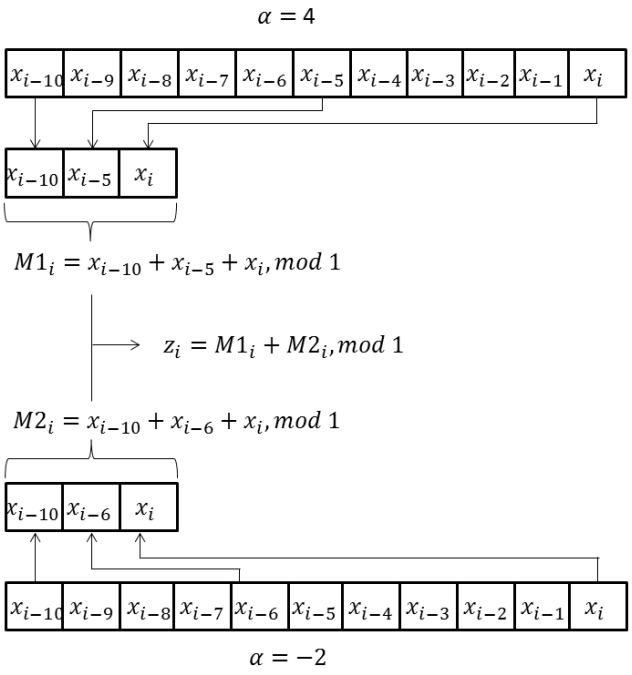


Figura 3.1: Diagrama de bloques del generador basado en retardos.

y tres unidades de memoria para cada una de las series de tiempo, de tal manera que los sistemas se iteran de forma independiente y con condiciones iniciales diferentes $x_{01} \neq x_{02}$, es importante señalar que con base a lo mostrado en la sección 2.2.2.1 el parámetro de bifurcación del mapeo logístico puede tomar valores negativos y además se propone tomar los valores extremos, con esto se tiene que el valor de $\alpha_1 = 4, \alpha_2 = -2$. En la figura 3.1 se muestra un diagrama de bloques del generador propuesto.

Por un lado tenemos la serie de tiempo $M1$ con parámetro de bifurcación $\alpha_1 = 4$, con esto aseguramos que el sistema tiene comportamiento caótico ya que presenta un valor positivo el exponente de Lyapunov. Si en este punto graficamos el espacio fase por medio de la serie de tiempo $M1$ generada por la ecuación (2.15), se puede observar la forma del mapeo logístico y es posible estimar el valor del parámetro de bifurcación, con el fin de eliminar esta posibilidad, la serie de tiempo $M1$ es generada por medio de sumar retardos de la misma serie de tiempo. Existe una gran cantidad de combinaciones que se pueden lograr por medio de los retardos, sin embargo cada retardo requiere de una unidad de memoria y tiempo de procesamiento, por esta razón buscamos combinaciones

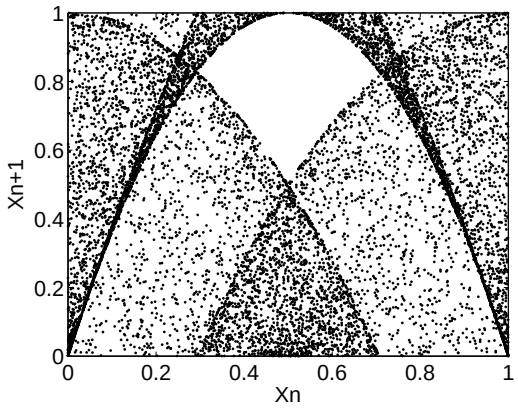


Figura 3.2: Espacio fase generado a partir de una serie de tiempo con una unidad de retardo.

que requieran la menor cantidad de retardos y la menor longitud en el retardo. Para el caso de usar dos retardos y sin importar la longitud de los retardos se puede observar en el espacio fase la forma del mapeo logístico acompañada de puntos como se observa en la figura 3.2, esta es la misma forma que se presenta en el trabajo de Kanso [42], donde se realiza la combinación de dos mapeos logísticos con el mismo parámetro de bifurcación y condiciones iniciales diferentes, hay que tomar en cuenta que los valores del mapeo logístico pertenecen al intervalo [0,1), para $\alpha = 4$. Por lo tanto después de realizar la suma de las dos series de tiempo se aplica la operación módulo 1 (mod 1). Por lo que para lograr que sea irreconocible la forma del mapeo logístico en el espacio fase se requiere de al menos la combinación de tres series de tiempo o bien el uso de 3 retardos. Por esta razón en nuestro algoritmo se requiere el valor de la iteración actual x_i, el valor de la iteración con un retardo de 5 unidades x_{i-5} y el valor de la iteración con un retardo de 10 unidades x_{i-10} hay que tener en cuenta que el uso de retardos contiguos genera patrones regulares en el espacio fase. Finalmente para limitar los valores de la serie $M1$ se utiliza la operación (mod 1) con esto garantizamos que $M1 \in [0,1)$, de tal forma que podemos expresar esta serie de la siguiente manera:

$$M1_i = x_{i-10} + x_{i-5} + x_i, \text{mod } 1. \tag{3.1}$$

Por otro lado, para producir la serie de tiempo $M2$ utilizamos un proceso similar pero ahora tomamos el parámetro de bifurcación $\alpha_2 = -2$. Esta serie de tiempo $M2$ está dada por la suma de la iteración actual x_i, la iteración con un retardo de seis unidades x_{i-6} y la

Tabla 3.1: Parte 1 de los resultados del banco de pruebas estadísticas.

Prueba estadística	Número de secuencias aprobadas	Número de secuencias falladas	Porcentaje de secuencias aprobadas
Frecuencia (Mono-bit)	1974	26	0.9870
Frecuencia dentro de un bloque (Bloque=128)	1974	26	0.9870
Corridas	1980	20	0.9900
Corrida larga	1981	19	0.9905
Rango de una matriz binaria	1978	22	0.9890
Transformada discreta de Fourier	1966	34	0.9834
Búsqueda de patrones sobrepuestos (Bloque=9)	1973	27	0.9865
Maurer	1977	23	0.9885
Entropía (Bloque=10)	1976	24	0.9880
Complejidad lineal (Bloque=500)	1980	20	0.9900

iteración con un retardo de 10 unidades x_{i-10}. De tal forma que esta serie se expresa de la siguiente manera:

$$M2_i = x_{i-10} + x_{i-6} + x_i, \text{mod } 1. \tag{3.2}$$

Hasta ahora tenemos $M1, M2 \in [0,1) \subset \Re$, en cada una de estas series se ha suprimido la forma del mapeo logístico en el espacio fase, sin embargo aún falta mezclar estas dos series de tiempo, para esto se sumarán las series y se aplicará nuevamente la operación (mod 1), quedando denotado este proceso como sigue:

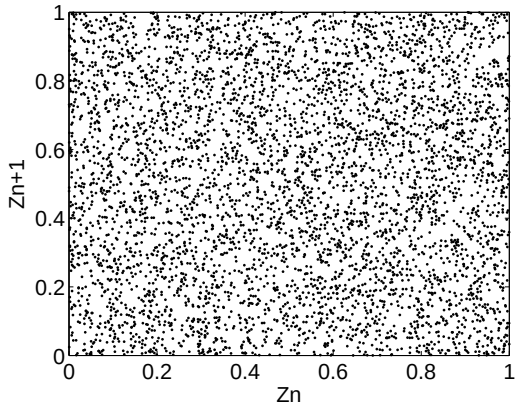


Figura 3.3: Espacio fase a partir de la serie de tiempo Z_i

$$Z_i = M1_i + M2_i, \text{ mod } 1. \tag{3.3}$$

Al observar el espacio fase generado a partir de Z_i obtenemos una nube de puntos como se observa en la figura 3.3, de esta forma es irreconocible el mapeo logístico en el espacio fase así la secuencia $Z_i \in [0,1) \subset \Re$. El siguiente paso es realizar una transformación para obtener una serie binaria $s(Z_i) \in \{0,1\}$, teniendo en mente que el número de ceros en la secuencia debe ser aproximadamente igual al número de unos, esta transformación se debe de realizar con la misma probabilidad como se muestra a continuación:

$$s_i = \begin{cases} 0, & 0 \leq Z_i \leq 0.5; \\ 1, & 0.5 < Z_i \leq 1. \end{cases} \tag{3.4}$$

Para evaluar este generador por medio de pruebas estadísticas se tomó una muestra de 2,000 secuencias con una longitud de 1,000,000 de elementos cada una. Los resultados de las pruebas se muestran en las tablas 3.1 y 3.2, en la primera columna se indica el nombre de la prueba, en la segunda y tercera columnas se indica el número de secuencias que pasaron y fallaron la prueba respectivamente, en la última columna se muestra la porción de secuencias que pasaron la prueba. Además en las figuras 3.4 y 3.5 se muestran los resultados junto con los intervalos de confianza, los detalles de las pruebas estadísticas se encuentran en el apéndice B.

Tabla 3.2: Parte 2 de los resultados del banco de pruebas estadísticas.

Prueba estadística	Número de secuencias aprobadas	Número de secuencias falladas	Porcentaje de secuencias aprobadas
Serial 1 (Bloque=16)	1975	25	0.9875
Serial 2 (Bloque=16)	1987	13	0.9935
Sumas acumulativas			
a)hacia adelante	1973	27	0.9865
b)hacia atrás	1987	13	0.9935
Búsqueda de patrones			
no sobrepuestos (Bloque=9)			
a)	1981	19	0.9905
b)	1984	16	0.9920
c)	1985	15	0.9925
d)	1985	15	0.9925
Caminatas aleatorias			
a)-4	1977	23	0.9886
b)-3	1974	26	0.9870
c)-2	1980	20	0.9902
d)-1	1974	26	0.9870
e)1	1972	28	0.9862
f)2	1983	17	0.9919
g)3	1977	23	0.9886
h)4	1988	12	0.9943
Caminatas aleatorias (Variante)			
a)-9	1985	15	0.9927
b)-8	1988	12	0.9943
c)-7	1991	9	0.9959
d)-6	1978	22	0.9894
e)-5	1977	23	0.9886
f)-4	1980	20	0.9902
g)-3	1977	23	0.9886
h)-2	1977	23	0.9886
i)-1	1985	15	0.9927
j)1	1988	12	0.9943
k)2	1988	12	0.9943
l)3	1987	13	0.9935
m)4	1893	17	0.9919
n)5	1980	20	0.9902
o)6	1980	20	0.9902
p)7	1978	22	0.9894
q)8	1978	22	0.9894
r)9	1985	15	0.9927

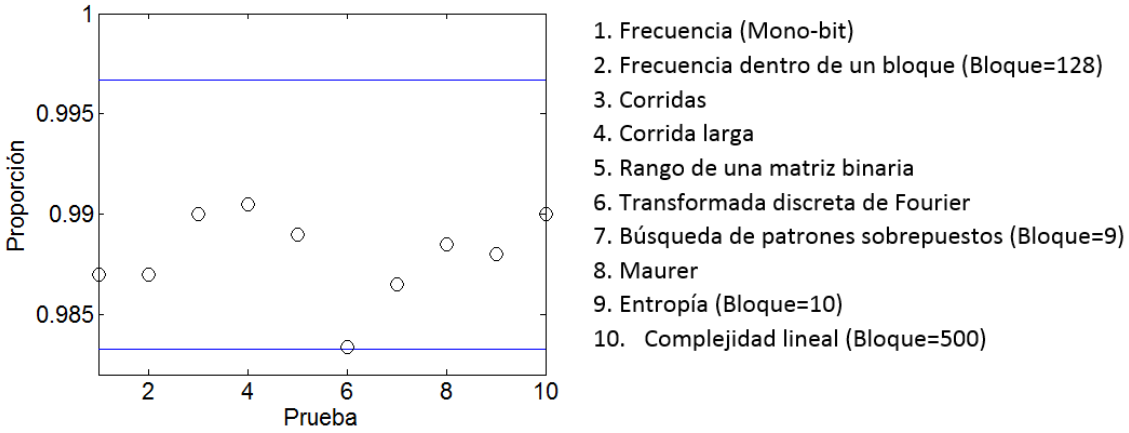


Figura 3.4: Parte 1 de los resultados estadísticos dentro del intervalo de confianza.

3.2. Generador pseudo-aleatorio basado en mapeos multi-modales

Como se mostró en el capítulo anterior los mapeos multi-modales son capaces de cambiar el número de modas que contiene el mapeo al variar el parámetro de bifurcación, esta propiedad es la base de este generador, ya que para lograr la combinación de varias series de tiempo, definimos solamente un mapeo k-modal, el cual contiene k series de tiempo, de esta forma garantizamos la combinación de múltiples series de tiempo y con solamente un mapeo. A continuación se muestra el algoritmo para cualquier valor de k y posteriormente se mostrará el algoritmo para el caso particular $k = 3$.

Paso 1: Definir el valor de $k \in \aleph$, y de igual forma se define el mapeo multi-modal basado en k modas

$$f_\beta = \beta(d_{r+1} - x)(x - d_r), x \in I_r. \tag{3.5}$$

En este paso se define la función f_β así como el máximo número de modas que podrá contener el mapeo.

Paso 2: Se calculan los valores de β_j para $j = 1, \ldots, k$ por medio de las siguientes ecuaciones

$$\beta_1 = (4)(k); \tag{3.6}$$

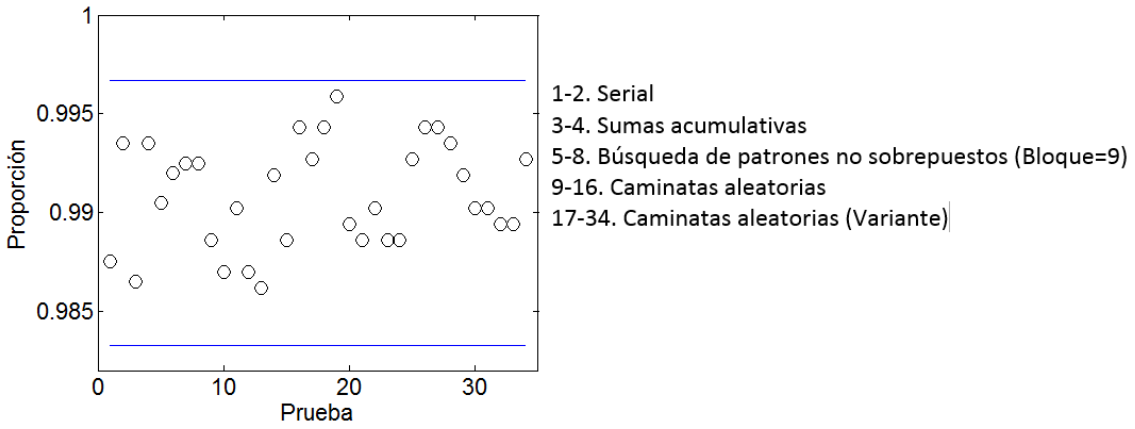


Figura 3.5: Parte 2 de los resultados estadísticos dentro del intervalo de confianza.

$$\beta_j = (j)(\beta_1);\ \text{para}\ j = 2,\ldots,k. \tag{3.7}$$

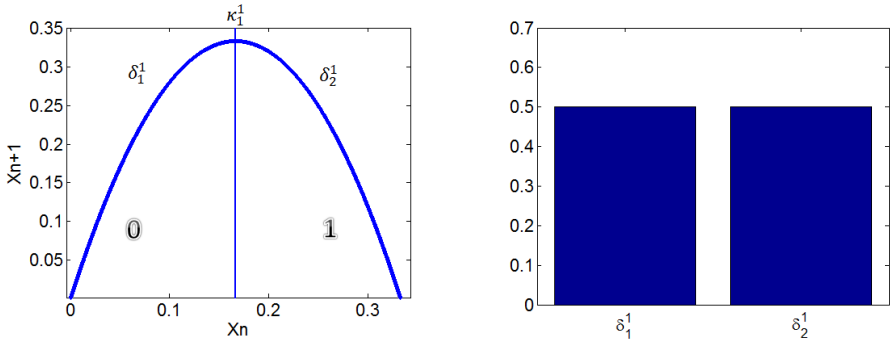
Se calculan k diferentes valores de parámetro de bifurcación β_j, en donde cada valor de β_j produce un mapeo con j modas. Cabe mencionar que al seleccionar de esta forma los valores del parámetro de bifurcación se evaden las ventanas periódicas y se garantizan órbitas con comportamiento caótico, por lo tanto el exponente de Lyapunov para β_j es positivo.

Paso 3: Tomar el valor de β_j y dividir el espacio fase en $2 * j$ intervalos denotados como $\delta_1^j,\ldots,\delta_{2*j}^j$ (figura 3.6), estas regiones están limitadas por los valores $\kappa_1^j,\ldots,\kappa_{(2*j)-1}^j$. Iterando el sistema $x_n^j = f(\beta_j, x_n)$ y asignando valores de cero y uno a los intervalos denotados por δ_1^1 y δ_2^1, respectivamente, con esto es posible generar la secuencia binaria ζ_n^j como sigue:

$$\zeta_n^{j=1} = \begin{cases} 0 \text{ si } x_n \in \delta_1^1 \\ 1 \text{ si } x_n \in \delta_2^1 \end{cases} \tag{3.8}$$

Es importante remarcar que el número de ceros es aproximadamente igual al número de unos en la secuencia, con una tolerancia del 1 %.

Paso 4: En este punto se han generado k series de tiempo caóticas mediante la función $x_n^j = f(\beta_j, x_n)$ y cada una de estas series produce una secuencia binaria ζ_n^j, por lo tanto se tienen k series binarias. Estas secuencias son mezcladas por medio de la operación XOR ($\oplus$) y de esta forma se obtiene la secuencia final Z.



(a) Espacio fase (b) Histograma

Figura 3.6: Espacio fase dividido en dos regiones δ_1^1, δ_2^1 junto a su histograma.

$$Z = \zeta^1 \oplus \zeta^2 \oplus \cdots \oplus \zeta^k \tag{3.9}$$

Para clarificar el funcionamiento del algoritmo anterior se muestra a continuación un ejemplo para el valor de $k = 3$. El primer paso es definir la función del mapeo con $k = 3$, para esto usamos la ecuación (3.5) y obtenemos el siguiente mapeo tri-modal:

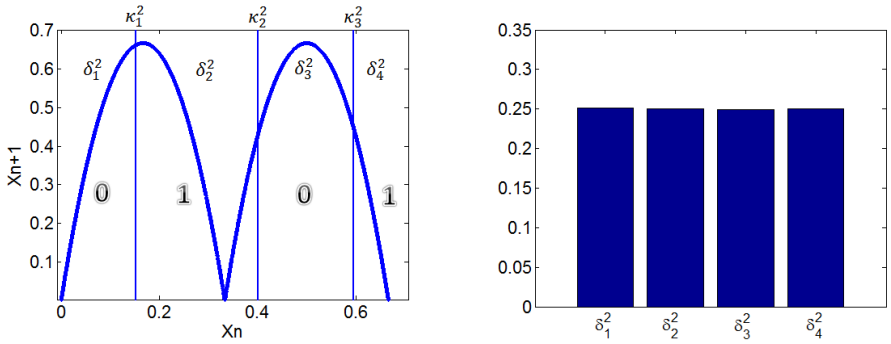
$$f_{\beta_j}(x) = \begin{cases} \beta_j(1/3 - x)x, & \text{para } x \in [0, 1/3); \\ \beta_j(2/3 - x)(x - 1/3), & \text{para } x \in [1/3, 2/3); \\ \beta_j(1 - x)(x - 2/3), & \text{para } x \in [2/3, 1]. \end{cases} \tag{3.10}$$

En el segundo paso del algoritmo encontramos los parámetros de bifurcación para generar 1, 2 y 3 modas, por lo que tenemos:

$$\beta_1 = 12; \beta_2 = 24; \beta_3 = 36;$$

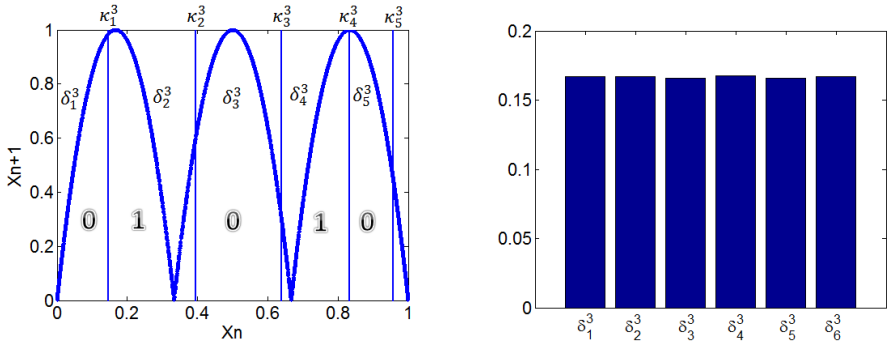
En el tercer paso tomamos el valor de β_1 para obtener la secuencia ζ^1, para esto el espacio fase es dividido en dos regiones δ_1^1 y δ_2^1, las cuales están divididas por $\kappa_1^1 = 1/6$. Estas dos regiones representan 0 y 1, respectivamente. Este proceso se muestra en la figura 3.6(a), al iterar el sistema se puede construir su histograma, el cual se muestra en la figura 3.6(b).

Este proceso se repite para el valor de β_2 al definir este valor ahora se tiene un mapeo bi-modal, para esto el espacio fase se divide en cuatro regiones $\delta_1^2, \delta_2^2, \delta_3^2, \delta_4^2$, estas regiones están limitadas por los valores $\kappa_1^2 = 0.1521, \kappa_2^2 = 0.40174, \kappa_3^2 = 0.5954$. En la



(a) Espacio fase (b) Histograma

Figura 3.7: Espacio fase dividido en cuatro regiones $\delta_1^2, \delta_2^2, \delta_3^2, \delta_4^2$ junto a su histograma.



(a) Espacio fase (b) Histograma

Figura 3.8: Espacio fase dividido en seis regiones $\delta_1^3, \delta_2^3, \delta_3^3, \delta_4^3, \delta_5^3, \delta_6^3$ junto a su histograma.

figura 3.7(a) se muestra el espacio fase del mapeo bi-modal así como las regiones δ_j^2 y los valores binarios que representa cada una de las regiones, en la figura 3.7(b) se muestra el histograma que se genera con estas particiones.

Por último, se toma el valor de β_3 el cual evoluciona a un mapeo tri-modal, ahora el espacio fase se divide en seis regiones $\delta_1^3, \delta_2^3, \delta_3^3, \delta_4^3, \delta_5^3, \delta_6^3$, a su vez estas regiones están limitadas por los valores $\kappa_1^3 = 0.1465, \kappa_2^3 = 0.396, \kappa_3^3 = 0.639, \kappa_4^3 = 0.8335, \kappa_5^3 = 0.9572$. En la figura 3.8(a) se muestra el espacio fase del mapeo tri-modal con sus respectivas regiones δ_j^3, en la figura 3.8(b) se muestra el histograma que se genera para este mapeo.

Este tipo de mapeos presenta una distribución estadística tipo U, sin embargo estamos interesados en obtener una distribución uniforme, esto es posible si dividimos el espacio fase en $2 * k$ regiones, de tal forma que al iterar el sistema se visite aproximadamente el

Tabla 3.3: Valores de κ para diferentes mapeos k-modales.

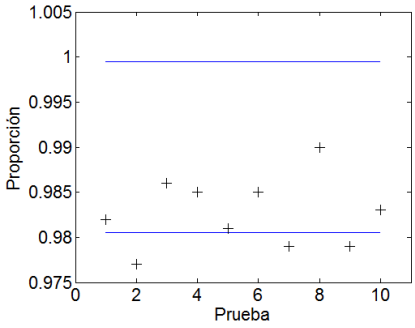
Mapeo	$\beta=\beta_1$	$\beta=\beta_2$	$\beta=\beta_3$	$\beta=\beta_4$
$k=1$	$\kappa_1^1=1/2$	-	-	-
$k=2$	$\kappa_1^1=1/4$	$\kappa_1^2=0.2273$ $\kappa_2^2=0.6020$ $\kappa_3^2=0.8931$	-	-
$k=3$	$\kappa_1^1=1/6$	$\kappa_1^2=0.1521$ $\kappa_2^2=0.4017$ $\kappa_3^2=0.5954$	$\kappa_1^3=0.1465$ $\kappa_2^3=0.3960$ $\kappa_3^3=0.6390$ $\kappa_4^3=0.8335$ $\kappa_5^3=0.9572$	-
$k=4$	$\kappa_1^1=1/8$	$\kappa_1^2=0.1140$ $\kappa_2^2=0.3012$ $\kappa_3^2=0.4465$	$\kappa_1^3=0.1105$ $\kappa_2^3=0.2970$ $\kappa_3^3=0.4790$ $\kappa_4^3=0.6250$ $\kappa_5^3=0.7180$	$\kappa_1^4=0.1070$ $\kappa_2^4=0.2911$ $\kappa_3^4=0.4829$ $\kappa_4^4=0.6576$ $\kappa_5^4=0.8034$ $\kappa_6^4=0.9114$ $\kappa_7^4=0.9776$

mismo número de veces cada región. El problema es definir los límites de cada región, en este caso los límites están definidos por los valores κ_i^j, los cuales se obtuvieron de forma experimental, es decir, se propone un límite inicial y en base a ese valor se obtiene la distribución, si el número de iteraciones en cada región es diferente en más del 1%, se realiza un nuevo ajuste del valor del límite κ_i^j, esto se hace de forma iterativa hasta encontrar un error menor al 1%.

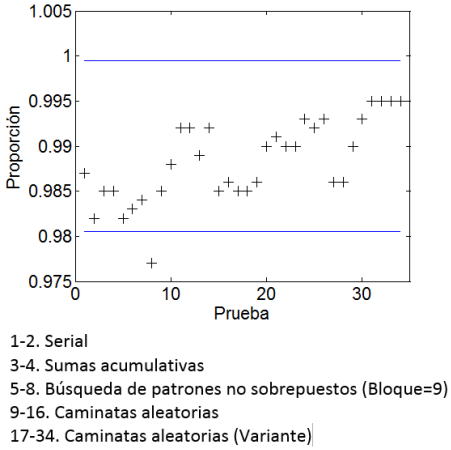
Por último, para obtener la salida del generador pseudo-aleatorio mezclamos las tres secuencias binarias $Z=\zeta^1\oplus\zeta^2\oplus\zeta^3$.

En la tabla 3.3 se muestran los valores que κ puede tomar para diferentes valores de $k=1,\ldots,4$.

Antes de evaluar las secuencias binarias con las pruebas estadísticas del NIST, se evalúa la correlación entre dos secuencias antes de ser binarizadas es decir en el dominio



(a) Parte 1

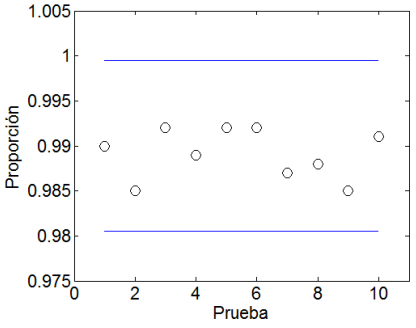


(b) Parte 2

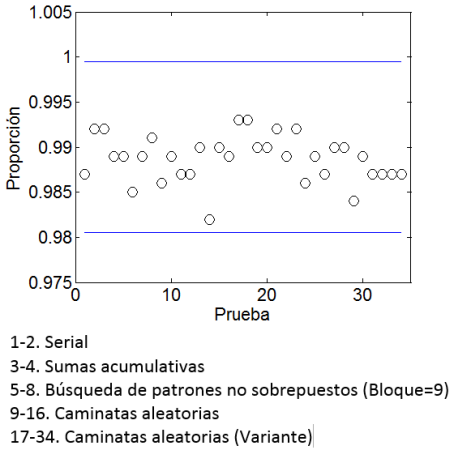
Figura 3.9: Resultados de las pruebas estadísticas del mapeo uni-modal dentro del intervalo de confianza.

Tabla 3.4: Coeficientes de correlación de las secuencias pesudo-aleatorias.

Uni-modal	$x = 0.24879652$	$x' = 0.248796520000001$	$C_{xx'} = 0.0015$
Bi-modal	$x = 0.19685672$	$x' = 0.196856720000001$	$C_{xx'} = 0.0018$
	$y = 0.86241957$	$y' = 0.862419570000001$	$C_{yy'} = -0.0022$
Tri-modal	$x = 0.38461795$	$x' = 0.384617950000001$	$C_{xx'} = -0.0004$
	$y = 0.73519846$	$y' = 0.735198460000001$	$C_{yy'} = -0.0013$
	$z = 0.48617852$	$z' = 0.486178520000001$	$C_{zz'} = -0.0016$
Cuatri-modal	$x = 0.26487146$	$x' = 0.264871460000001$	$C_{xx'} = -0.0006$
	$y = 0.52975314$	$y' = 0.529753140000001$	$C_{yy'} = 0.00001$
	$z = 0.93457812$	$z' = 0.934578120000001$	$C_{zz'} = -0.0004$
	$w = 0.63789514$	$w' = 0.637895140000001$	$C_{ww'} = 0.0008$



(a) Parte 1



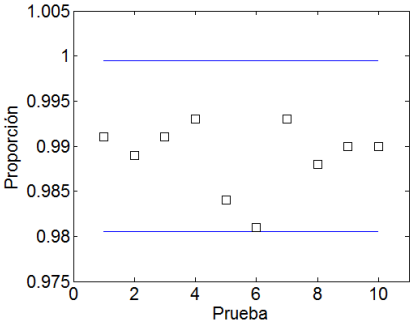
(b) Parte 2

Figura 3.10: Resultados de las pruebas estadísticas del mapeo bi-modal dentro del intervalo de confianza.

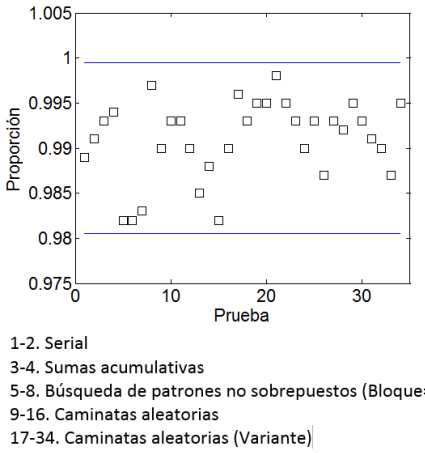
de los reales, estas secuencias son generadas con condiciones iniciales cercanas, además podemos verificar la sensibilidad a las condiciones iniciales. El coeficiente de correlación se denota como C_{xy}, para esto se requiere de un par de secuencias $x = [x_1, \ldots, x_N]$, $y = [y_1, \ldots, y_N]$.

$$C_{xy} = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{[\sum_{i=1}^{N}(x_i - \bar{x})^2]^{1/2}[\sum_{i=1}^{N}(y_i - \bar{y})^2]^{1/2}} \tag{3.11}$$

En donde $\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$ y $\bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i$ son los valores promedio de cada una de las secuencias. Para el caso de un mapeo uni-modal $k = 1$ se tiene una condición inicial (x_{01}) y una serie de tiempo (x), para construir la serie de tiempo (y) se tomará como condición inicial $x_{02} = x_{01} + \delta$. Para el caso del mapeo bi-modal $k = 2$ se producen dos secuencias x_n^1, x_n^2 con dos condiciones iniciales x_{01}, x_{02}, para evaluar los coeficientes de correlación se requieren dos secuencias más $x_n'^1, x_n'^2$ con dos condiciones iniciales cercanas $x_{01} + \delta, x_{02} + \delta$. Este proceso lo realizamos hasta el mapeo cuatri-modal, los resultados se muestran en la tabla 3.4 en donde es posible observar que no existe correlación entre las series generadas, como consecuencia de forma implícita se puede verificar que se tiene

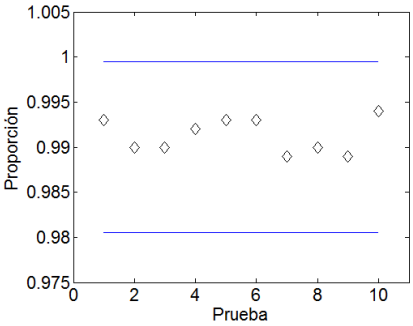


(a) Parte 1

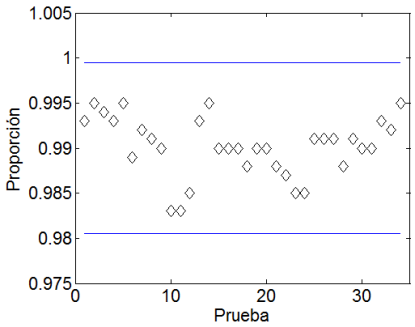


(b) Parte 2

Figura 3.11: Resultados de las pruebas estadísticas del mapeo tri-modal dentro del intervalo de confianza.



(a) Parte 1



(b) Parte 2

Figura 3.12: Resultados de las pruebas estadísticas del mapeo cuatri-modal dentro del intervalo de confianza.

Tabla 3.5: Parte 1 de los resultados del banco de pruebas estadísticas.

Prueba estadística	Uni-modal Porción de secuencias aprobadas	Bi-modal Porción de secuencias aprobadas	Tri-modal Porción de secuencias aprobadas	Cuatri-modal Porción de secuencias aprobadas
Frecuencia (Mono-bit)	0.9820	0.9900	0.9910	0.9930
Frecuencia dentro de un bloque (Bloque=128)	0.9770*	0.9850	0.9890	0.9900
Corridas	0.9860	0.9920	0.9910	0.9900
Corrida larga	0.9850	0.9890	0.9930	0.9920
Rango de una matriz binaria	0.9810	0.9920	0.9840	0.9930
Transformada discreta de Fourier	0.9850	0.9920	0.9840	0.9930
Búsqueda de patrones sobrepuestos (Bloque=9)	0.9790*	0.9870	0.9930	0.9890
Maurer	0.9900	0.9880	0.9880	0.9900
Entropía (Bloque=10)	0.9790*	0.9850	0.9900	0.9890
Complejidad lineal (Bloque=500)	0.9830	0.9910	0.9900	0.9940

Tabla 3.6: Parte 2 de los resultados del banco de pruebas estadísticas.

Prueba estadística	Uni-modal Secuencias aprobadas	Bi-modal Secuencias aprobadas	Tri-modal Secuencias aprobadas	Cuatri-modal Secuencias aprobadas
Serial 1 (Bloque=16)	0.9870	0.9870	0.9890	0.9930
Serial 2 (Bloque=16)	0.9820	0.9920	0.9910	0.9950
Sumas acumulativas				
a)Hacia adelante	0.9850	0.9920	0.9930	0.9940
b)Hacia atrás	0.9850	0.9890	0.9940	0.9930
Búsqueda de patrones no sobrepuestos (Bloque=9)				
a)	0.9820	0.9890	0.9820	0.9950
b)	0.9830	0.9850	0.9820	0.9890
c)	0.9840	0.9890	0.9830	0.9920
d)	0.9770*	0.9910	0.9970	0.9910
Caminatas aleatorias				
a) -4	0.9850	0.9860	0.9900	0.9900
b) -3	0.9880	0.9890	0.9930	0.9830
c) -2	0.9920	0.9870	0.9930	0.9830
d) -1	0.9920	0.9870	0.9900	0.9850
e) 1	0.9890	0.9900	0.9850	0.9930
f) 2	0.9920	0.9820	0.9880	0.9950
g) 3	0.9850	0.9900	0.9820	0.9900
h) 4	0.9860	0.9890	0.9900	0.9900
Caminatas aleatorias (Variante)				
a) -9	0.9850	0.9930	0.9960	0.9900
b) -8	0.9850	0.9930	0.9930	0.9880
c) -7	0.9860	0.9900	0.9950	0.9900
d) -6	0.9900	0.9900	0.9950	0.9900
e) -5	0.9910	0.9920	0.9980	0.9880
f) -4	0.9900	0.9890	0.9950	0.9870
g) -3	0.9900	0.9920	0.9930	0.9850
h) -2	0.9930	0.9860	0.9900	0.9850
i) -1	0.9920	0.9890	0.9930	0.9910
j) 1	0.9930	0.9870	0.9870	0.9910
k) 2	0.9860	0.9900	0.9930	0.9910
l) 3	0.9860	0.9900	0.9920	0.9880
m) 4	0.9900	0.9840	0.9950	0.9910
n) 5	0.9930	0.9890	0.9930	0.9900
o) 6	0.9950	0.9870	0.9910	0.9900
p) 7	0.9950	0.9870	0.9900	0.9930
q) 8	0.9950	0.9870	0.9870	0.9920
r) 9	0.9950	0.9870	0.9950	0.9950

sensibilidad a pequeños cambios en las condiciones iniciales.

Para caracterizar este generador además del coeficiente de correlación se utilizaron las pruebas estadísticas del NIST para esto se tomó una muestra de 1,000 secuencias con una longitud de 1,000,000 de elementos cada una. Los resultados para diferentes mapeos se muestran en las tablas 3.5 y 3.6, se puede observar un * en aquellas pruebas en donde las secuencias no obtuvieron resultados satisfactorios. Además se pueden observar los resultados en forma gráfica junto al intervalo de confianza para cada uno de los mapeos, en la figura 3.9 se muestran los resultados para el mapeo logístico $k = 1$, en la figura 3.10 se muestran los resultados del mapeo bi-modal $k = 2$, en la figura 3.11 se muestran los resultados del mapeo tri-modal $k = 3$, y por último en la figura 3.12 se muestran los resultados del mapeo cuatri-modal $k = 4$.

En el apéndice B se muestran brevemente las pruebas estadísticas contenidas en el banco de pruebas del NIST así como la interpretación de resultados para la obtención de los valores del intervalo de confianza.

Capítulo 4

Cifrado de imágenes

En este capítulo se propone una función para el cifrado de información en flujo, cabe recordar que la parte central de estos cifrados son las secuencias pseudo-aleatorias, por lo que partiremos de la suposición que el generador en cuestión entrega resultados satisfactorios en las pruebas de aleatoriedad. Para esto se mostrará el funcionamiento y análisis estadístico de tres funciones de cifrado en flujo: la primera función de cifrado es la más sencilla y se basa solamente en el cifrado de Vernam, posteriormente mostraremos una función de cifrado con una pequeña modificación usada en [104], y por último se mostrará la función de cifrado en flujo que se propone.

En el caso de las secuencias pseudo-aleatorias se utilizaron las pruebas estadísticas propuestas por el NIST para caracterizar las secuencias y decidir si el generador puede ser usado en el área de criptografía. Para el caso de la función de cifrado se realiza de igual forma un análisis estadístico por medio de pruebas de seguridad, estas pruebas se realizan sobre la información cifrada y lo que se busca es medir la aleatoriedad que produce la función.

Los cifrados como el 3DES o el AES están diseñados para cifrar texto, por lo que cada caracter se representa por su código ASCII, si usamos estos cifrados con imágenes se requiere de mayor capacidad de cómputo y mayor cantidad de tiempo si lo comparamos cuando se cifra texto. Este trabajo de tesis se enfoca en cifrar imágenes en escala de grises por lo que cada pixel se representa por una cadena de 8 bits teniendo un total 256 valores diferentes.

Por último además de las pruebas estadísticas de seguridad se aplicarán dos ataques los cuales son los más comunes.

4.1. Pruebas estadísticas

En esta sección presentaremos las pruebas de seguridad que se utilizarán para evaluar las funciones de cifrado al utilizar imágenes.

- Correlación de pixeles

En general uno de los principales problemas al cifrar imágenes se debe a que los valores de los pixeles pueden estar altamente correlacionados en cualquier dirección (vertical, horizontal y diagonal). Sin embargo la función de cifrado debe ser capaz de producir cambios en los valores de todos los pixeles y de esta forma obtener baja correlación en pixeles adyacentes. Para lograr este objetivo se debe obtener una distribución uniforme de pixeles en la imagen cifrada sin importar la distribución de la imagen en la entrada, esto es precisamente lo que se define como concepto de confusión.

Para poder cuantificar y comparar la correlación de pixeles adyacentes en diferentes direcciones y en diferentes imágenes, se define el coeficiente de correlación r_{xy} el cual está dado por la covarianza de dos pixeles entre el producto de la desviación estándar de cada uno de los pixeles, como se muestra a continuación:

$$r_{xy} = \frac{cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}}, \tag{4.1}$$

$$E(x) = \frac{1}{\eta}\sum_{i=1}^{\eta} x_i, \tag{4.2}$$

$$D(x) = \frac{1}{\eta}\sum_{i=1}^{\eta} (x_i - E(x))^2, \tag{4.3}$$

$$cov(x,y) = \frac{1}{\eta}\sum_{i=1}^{\eta} (x_i - E(x))(y_i - E(y)), \tag{4.4}$$

donde x,y representan dos pixeles adyacentes y η es el número total de duplas (x,y) en este trabajo tomamos $\eta = 2000$. Se espera que la imagen original tenga un coeficiente de correlación, cercano a 1, el cual indica que los pixeles adyacentes tienen valores muy similares, por otro lado se espera que la imagen cifrada tenga un coeficiente de correlación cercano a 0, indicando que los valores de pixeles adyacentes son diferentes.

- Entropía

La entropía es una medida de aleatoriedad la cual puede ser usada para caracterizar la textura de una imagen, la entropía fue definida por Shannon [4] y se expresa de la siguiente forma:

$$H(s) = -\sum_{i=0}^{2^n-1} P(s_i) log_2 P(s_i), \tag{4.5}$$

La información puede ser vista como una secuencia de símbolos aleatorios, los cuales forman parte de un alfabeto finito. El alfabeto que se utiliza para representar una imagen en escalas de grises está dado por 8 bits, esto es, para representar un pixel se requieren 8 bits. Así tenemos 2^8 símbolos que conforman el alfabeto $S = \{s_0, s_1, \ldots, s_{255}\}$ y además podemos asociar una probabilidad a cada elemento $\{P(s_0), P(s_1), \ldots, P(s_{255})\}$.

La cantidad de información está relacionada con la ocurrencia o frecuencia de cada símbolo. Un símbolo con poca probabilidad es el que provee de mayor información. Por lo tanto para maximizar el valor de la entropía todos los elementos deben tener la misma probabilidad.

- Calidad de cifrado

Un buen proceso de cifrado crea grandes cambios en el valor de los pixeles, donde estos nuevos valores de los pixeles deben de ser completamente diferentes a los de la imagen original. También se requiere que estos cambios sean irregulares así entre más cambios se encuentren en los valores de los pixeles más efectivo será el algoritmo de cifrado y por lo tanto tendrá una mejor calidad. La calidad de cifrado fue definida en [174] y se representa como el promedio de los cambios de cada nivel en la escala de grises:

$$Q = \frac{\sum_{L=0}^{255} |H_L(C) - H_L(P)|}{256}, \tag{4.6}$$

Donde L es el nivel de la escala de grises, $H_L(C)$ y $H_L(P)$ son el número de repeticiones de cada valor de gris en la imagen cifrada y en la imagen original respectivamente.

- NPCR, UACI

Existen dos medidas estadísticas que pueden determinar si un cifrado resiste ataques de tipo diferencial, en este ataque típicamente el oponente realiza una pequeña modificación en un pixel de la imagen original y a su vez observa los cambios realizados en la

imagen cifrada, tratando de encontrar una relación entre las dos imágenes. Si un cambio pequeño en la imagen original produce un cambio significativo en la imagen cifrada este ataque se vuelve ineficiente y prácticamente inútil.

El primer criterio para evaluar estos cambios se conoce como: La razón de cambio de pixeles (Number of Pixel Change Rate) y se define de la siguiente forma:

$$NPCR = \frac{\sum_{i,j} \delta(i,j)}{\nu} \times 100\,\%; \tag{4.7}$$

$$\delta(i,j) = \begin{cases} 0 & \text{si } C_1(i,j) = C_2(i,j); \\ 1 & \text{si } C_1(i,j) \neq C_2(i,j); \end{cases} \tag{4.8}$$

donde ν es el número total de pixeles en la imagen, C_1 es la imagen cifrada de P_1, C_2 es la imagen cifrada de P_2. Para el cálculo de la razón de cambio de pixeles se considera a P_1 y P_2 iguales a excepción de un solo pixel y además C_1, C_2 son cifradas usando la misma llave.

El segundo criterio se le conoce como el promedio del cambio de la intensidad (Unified Average Changing Intensity) y se define como:

$$UACI = \frac{1}{\nu}\left[\sum_{i,j} \frac{|C_1(i,j) - C_2(i,j)|}{2^8 - 1}\right]. \tag{4.9}$$

Es importante mencionar que para el correcto funcionamiento de estas métricas, el pixel a modificar debe de ser el primer pixel de la imagen, de otra forma, si se modifica el último pixel de la imagen no se presenta ningún cambio significativo y por lo tanto estas medidas son nulas.

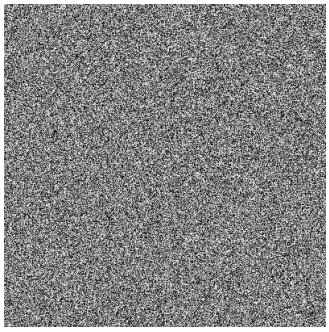
4.2. Función de cifrado basada en operación XOR

En este algoritmo se toman las funciones de cifrado y descifrado propuestas por Vernam las cuales se mostraron en la sección 2.1.1

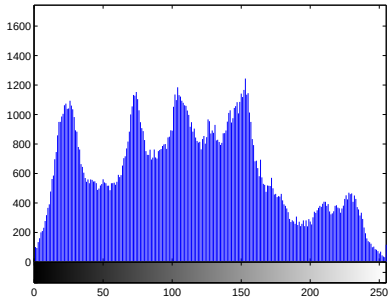
$$\begin{aligned} C_i &= e(P_i) = P_i \oplus s_i, \\ P_i &= d(C_i) = C_i \oplus s_i, \end{aligned} \tag{4.10}$$

donde P_i es el i-ésimo pixel (8 bits) de la imagen original, C_i es el i-ésimo pixel de la imagen cifrada y s_i es la secuencia pseudo-aleatoria en grupos de 8 bits. Partimos del

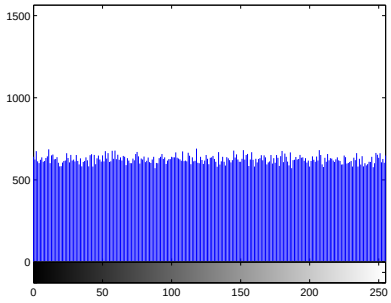
(a) Lenna, imagen original *P*.



(b) Imagen cifrada *C*.



(c) Distribución de pixeles de la imagen *P*.



(d) Distribución de pixeles de la imagen *C*.

Figura 4.1: Cifrado de imagen por medio de la operación XOR, usando el generador basado en series de tiempo con retardo.

hecho de que la secuencia pseudo-aleatoria a usarse ha pasado las pruebas de aleatoriedad del NIST. A continuación se mostrará el funcionamiento y análisis de los resultados de las pruebas de seguridad y para ello se usará el generador pseudo-aleatorio basado en series de tiempo con retardo mostrado en la sección 3.1.

Primeramente se crea una secuencia de bits con los parámetros $\alpha_1 = 4, \alpha_2 = -2, x_{01} = 0.347851763254826, x_{02} = 0.865484627895467$, así la llave del criptosistema es la combinación de las dos condiciones iniciales, de tal forma que para descifrar la información se requiere crear la misma secuencia de bits y esto se logra por medio de las condiciones iniciales. Una vez que se tiene la secuencia de bits la imagen se cifra por medio de la ecuación (4.10), en la figura 4.1 se muestra la imagen original *P* y la imagen cifrada *C* junto a sus correspondientes histogramas.

A continuación se muestra el análisis estadístico de la imagen cifrada por medio de la

Tabla 4.1: Correlación de pixeles adyacentes en las imágenes *P* y *C* en distintas direcciones.

Dirección	Imagen original	Imagen cifrada
Vertical	0.9844	0.0288
Horizontal	0.9661	-0.0391
Diagonal	0.9534	0.0059

Tabla 4.2: Entropía de las imágenes *P* y *C*.

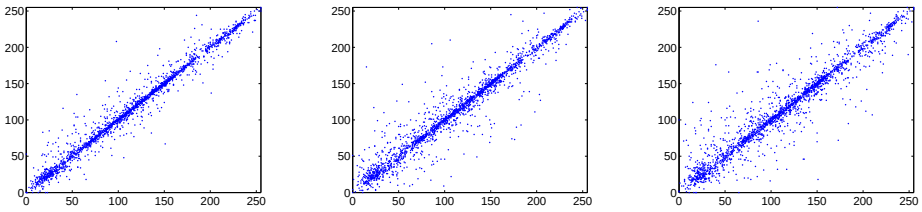
Entropía	Imagen original	Imagen cifrada
Lenna	7.8059	7.9989

ecuación (4.10). En la tabla 4.1 se muestra la correlación de pixeles adyacentes calculados por medio de la ecuación (4.1) en dirección vertical, horizontal y diagonal de las imágenes *P* y *C*.

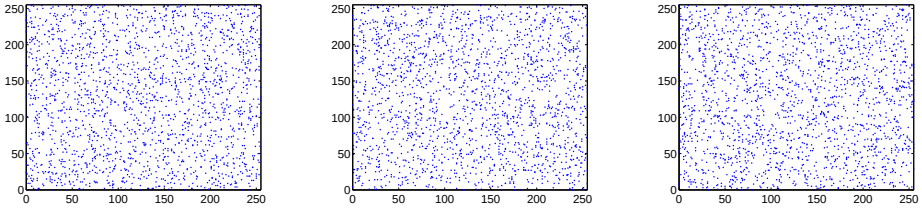
A partir de los resultados se puede observar que en la imagen *P* existe una gran correlación de pixeles en cualquier dirección mientras que en la imagen *C* los pixeles adyacentes tienen una baja correlación. En la figura 4.2 se muestra el diagrama de dispersión de pixeles de las imágenes *P* y *C*, de igual forma se puede observar que para la imagen *P* el conjunto de puntos tienden a formar una línea recta, mientras que en la imagen *C* se forma una nube de puntos en la que no existe algún tipo de correlación.

Por otro lado, podemos calcular la entropía usando la ecuación (4.5), en la tabla 4.2 se muestran los resultados de la entropía para las imágenes *P* y *C*. Se puede observar que en la imagen *C* el valor de la entropía es mayor que en *P* esto es debido a que la probabilidad de ocurrencia de cada nivel de la escala de grises tiende a ser uniforme. La entropía ideal o máxima en una imagen cifrada es de 8 bits ya que para representar cada pixel se requiere de 8 bits.

Por último, obtendremos las medidas NPCR y UACI, para esto se requiere de dos imágenes P_1, P_2 que solo difieran en un pixel y sus correspondientes imágenes cifradas C_1, C_2, pero cifradas con la misma llave. Los coeficientes calculados se encuentran en la tabla 4.3 mientras que en la figura 4.3 se muestran dichas imágenes.



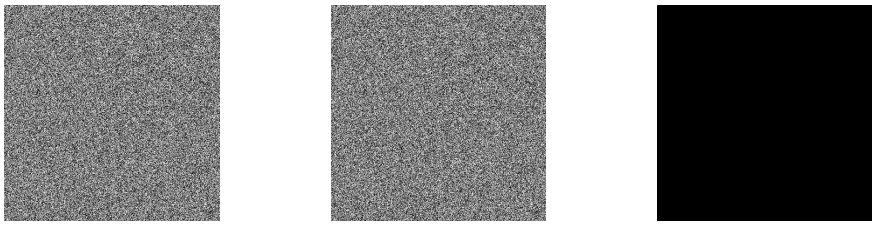
(a) Dirección vertical P. (b) Dirección horizontal P. (c) Dirección diagonal P.



(d) Dirección vertical C. (e) Dirección horizontal C. (f) Dirección diagonal C.

Figura 4.2: Diagramas de dispersión de pixeles adyacentes en diferentes direcciones de las imágenes P y C.

(a) Imagen P_1. (b) Imagen P_2. (c) Diferencia de las imágenes P_1, P_2.



(d) Imagen C_1. (e) Imagen C_2. (f) Diferencia de las imágenes C_1, C_2.

Figura 4.3: Cifrado de dos imágenes que son idénticas excepto en un pixel.

Tabla 4.3: NPCR y UACI de las imágenes C_1, C_2.

	Cifrado XOR
NPCR	6.2500e-04 %
UACI	3.3824e-04 %

Se puede observar que estas medidas presentan niveles muy bajos, esto es debido a que los pixeles se cifran de forma independiente. En la figura 4.3 (c) y (f) se observa la diferencia entre P_1, P_2 y C_1, C_2, la diferencia entre estas imágenes es cero excepto en un pixel. Recordemos que el concepto de difusión nos indica que cualquier modificación por muy pequeña que sea se transforma en un gran cambio en la salida. Como podemos ver en la figura 4.3 (f) este cifrado no cumple con el concepto de difusión y además podemos utilizar esta vulnerabilidad para ejecutar un ataque diferencial.

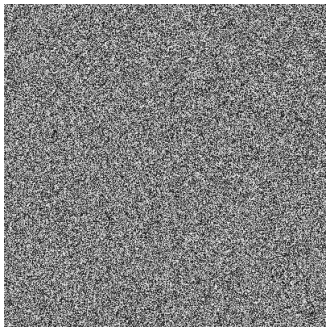
4.3. Función de cifrado basada en operación XOR y retardo

Para este algoritmo se toman las funciones de cifrado y descifrado propuestas por Vernam y además añadimos un elemento a la función de cifrado, este elemento es un retardo, se agrega con la finalidad de cumplir el concepto de difusión y de esta forma evitar ataques de tipo diferencial. El algoritmo queda dado de la siguiente forma:

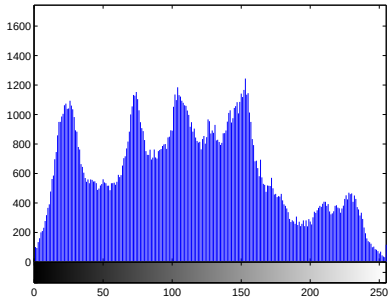
$$\begin{aligned} C_1 &= e(P_1) = P_1 \oplus s_1 \oplus IV; \\ C_i &= e(P_i) = P_i \oplus s_i \oplus C_{i-1}; \\ \\ P_1 &= d(C_1) = C_1 \oplus s_1 \oplus IV; \\ P_i &= d(C_i) = C_i \oplus s_i \oplus C_{i-1}; \end{aligned} \tag{4.11}$$

donde P_i es el i-ésimo pixel de la imagen original, C_i es el i-ésimo pixel de la imagen cifrada, IV es un vector de incialización $IV \in (0, 255)$ y s_i es la secuencia pseudo-aleatoria en grupos de 8 bits. De igual forma que en el cifrado anterior suponemos que la secuencia a utilizar s_i pasa las pruebas de aleatoriedad, por lo que tomamos nuevamente el generador presentado en la sección 3.1.

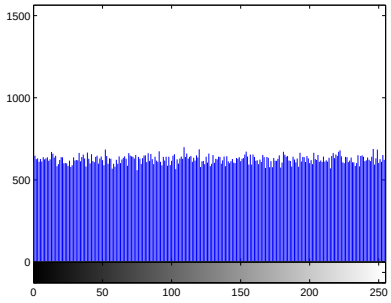
(a) Lenna, imagen original P.



(b) Imagen cifrada C.



(c) Distribución de pixeles de la imagen P.



(d) Distribución de pixeles de la imagen C.

Figura 4.4: Cifrado de imagen por medio de la operación XOR con retardo, usando el generador basado en series de tiempo con retardo.

Creamos la secuencia pseudo-aleatoria con los mismos parámetros $\alpha_1 = 4$, $\alpha_2 = -2$, $x_{01} = 0.347851763254826$, $x_{02} = 0.865484627895467$, de tal forma que el único cambio se encuentra en la función de cifrado dado por la ecuación (4.11), recordemos que la secuencia s_i es el núcleo de los cifrados en flujo, sin embargo dependiendo del uso que se le dé, por medio de la función de cifrado tendremos cambios en la información cifrada. En la figura 4.4 se muestra la imagen original P y la imagen cifrada C.

A continuación se muestran los resultados de las pruebas de seguridad aplicadas a la imagen cifrada. En la tabla 4.4 se muestra la correlación de pixeles adyacentes de las imágenes P y C. Se puede observar que en la imagen P existe una gran correlación de pixeles en cualquier dirección, y por otro lado en la imagen C los pixeles adyacentes presentan baja correlación. En la figura 4.5 se muestra el diagrama de dispersión de pixeles de las imágenes P y C en diferentes direcciones, se puede observar que en la imagen P

Tabla 4.4: Correlación de pixeles adyacentes en las imágenes P y C en distintas direcciones.

Dirección	Imagen original	Imagen cifrada
Vertical	0.9844	0.0130
Horizontal	0.9661	0.0143
Diagonal	0.9534	-0.0111

Tabla 4.5: Entropía de las imágenes P y C.

Entropía	Imagen original	Imagen cifrada
Lenna	7.8059	7.9988

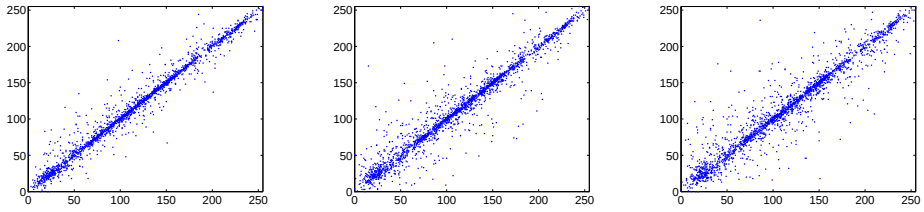
el conjunto de puntos tienden a formar una línea recta, mientras que en la imagen C se forma una nube de puntos indicando que no existe ningún tipo de correlación.

La entropía la podemos calcular usando la ecuación (4.5), en la tabla 4.5 se muestra la entropía para las imágenes P,C.

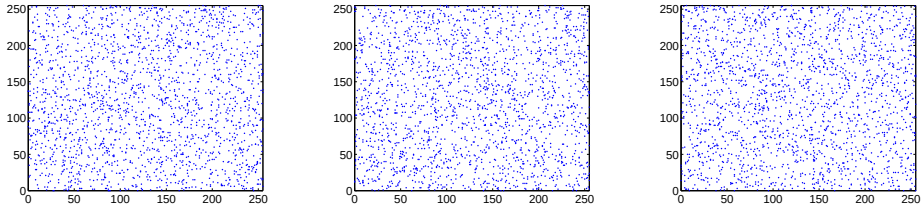
Por último, obtendremos los coeficientes NPCR y UACI, con estas medidas podemos medir la sensibilidad que presenta la función de cifrado a pequeños cambios para esto se requiere de dos imágenes P_1 y P_2 que solo difieran en un pixel χ y sus correspondientes imágenes cifradas C_1, C_2 utilizando la misma llave.

Para el cifrado presentado en la sección anterior se mostró que los pixeles se cifran de forma independiente, con esto podemos decir que sin importar el valor del pixel χ, en la imagen cifrada solo se modificará un pixel, sin embargo para esta función de cifrado no podemos hacer la misma aseveración, por lo tanto, el pixel χ puede tomar 256 valores diferentes y debemos calcular los coeficientes NPCR y UACI para cada posible valor, finalmente calcularemos el promedio de cada medida estadística. En la figura 4.6 se muestran los coeficientes NPCR y UACI obtenidos al hacer el barrido de todos los posibles valores para el pixel χ, y en la tabla 4.6 se muestra el promedio de dichos valores.

Se puede observar que estas medidas presentan niveles muy altos, esto es debido a que el retardo que se agregó en la función de cifrado crea un efecto avalancha, esto es, cualquier cambio en el valor del pixel χ se difunde por toda la imagen logrando crear

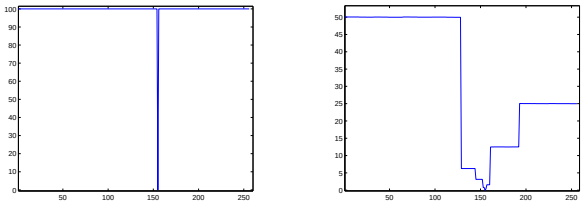


(a) Dirección vertical P. (b) Dirección horizontal P. (c) Dirección diagonal P.



(d) Dirección vertical C. (e) Dirección horizontal C. (f) Dirección diagonal C.

Figura 4.5: Diagramas de dispersión de pixeles adyacentes en diferentes direcciones de las imágenes P y C.



(a) NPCR. (b) UACI.

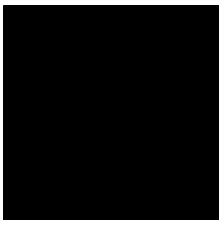
Figura 4.6: Coeficiente NPCR y UACI para cada posible valor de χ.

Tabla 4.6: NPCR y UACI de las imágenes C_1, C_2.

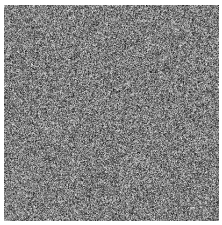
	Cifrado con retardo
NPCR	99.6094 %
UACI	33.3400 %

(a) Imagen P_1.

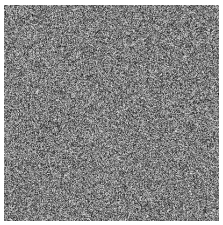
(b) Imagen P_2.



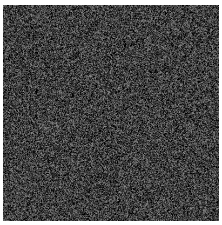
(c) Diferencia de las imágenes P_1, P_2.



(d) Imagen C_1.



(e) Imagen C_2.



(f) Diferencia de las imágenes C_1, C_2.

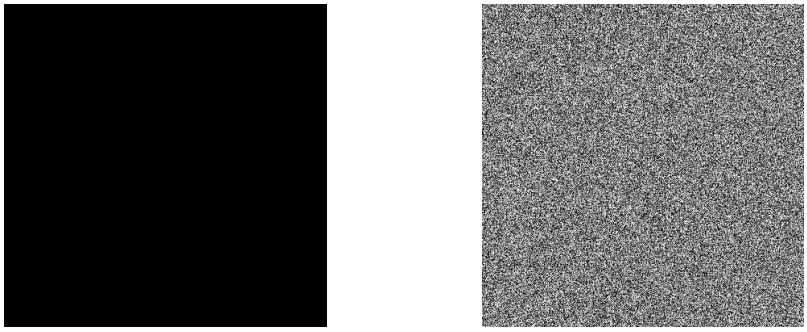
Figura 4.7: Cifrado de dos imágenes que son idénticas excepto en un pixel.

imágenes cifradas diferentes cuando se presentan cambios muy pequeños, esto es precisamente lo que indica el concepto de difusión, cualquier modificación por muy pequeña que sea se transforma en un gran cambio en la salida.

En la figura 4.7 (c) y (f) se observa la diferencia entre P_1, P_2 y C_1, C_2, la diferencia entre las imágenes P_1 y P_2 es cero excepto en un pixel, sin embargo al realizar la diferencia entre las imágenes C_1 y C_2 se puede observar que no hay pixeles idénticos esto lo muestra el coeficiente NPCR y la diferencia en la intensidad de pixeles la muestra el coeficiente UACI.

Con esta propiedad la función de cifrado se vuelve resistente a los ataques de tipo diferencial precisamente por la sensibilidad a pequeños cambios, sin embargo el cifrado es vulnerable a los ataques de texto escogido, esto es debido a que la función de cifrado si bien difunde pequeños cambios por toda la imagen siempre difundirá el mismo cambio debido a que la función es determinista, dicho de otra forma los cambios que se producen en la salida serán los mismos cada vez que se produzca el mismo cambio en la entrada.

Para ejemplificar el funcionamiento del ataque de texto escogido supongamos que en la imagen P todos los valores de los pixeles son cero, es decir la imagen es completa-



(a) (b)

Figura 4.8: Cifrado de imagen P en donde todos sus valores son cero.

mente negra, además supongamos que el vector de inicialización $IV = 0$ y la imagen es cifrada por medio de una llave k_1 lo que obtenemos es la imagen C. En la figura 4.8 se muestran dichas imágenes. Si analizamos a detalle la imagen cifrada podemos observar las siguientes características.

Para cifrar el primer pixel realizamos la siguiente operación

$$C_1 = P_1 \oplus s_1 \oplus IV. \tag{4.12}$$

Si tenemos en cuenta que $P_1 = 0$ y $IV = 0$, el primer pixel cifrado lo podemos obtener de la siguiente forma

$$C_1 = 0 \oplus s_1 \oplus 0 = s_1. \tag{4.13}$$

Por lo tanto, lo que está representado en el primer pixel de la imagen cifrada son los primeros ocho bits de la secuencia pseudo-aleatoria, los cuales fueron expuestos sin necesidad de conocer la llave k_1.

Si aplicamos el mismo procedimiento para el siguiente pixel se tiene lo que se muestra a continuación:

$$\begin{aligned} C_2 &= P_2 \oplus s_2 \oplus C_1; \\ C_2 &= 0 \oplus s_2 \oplus s_1; \\ C_2 &= s_2 \oplus s_1. \end{aligned} \tag{4.14}$$

Por lo tanto, lo que está representado en el segundo pixel de la imagen cifrada es el

resultado de la operación XOR entre s_2 y s_1, dado que conocemos el valor de C_2 y de s_1 podemos obtener el valor de s_2. Así al realizar este procedimiento de forma iterativa se puede obtener la secuencia s_i.

De forma general podemos obtener la secuencia pseudo-aleatoria sin conocer el valor de la llave k_1, lo único que se requiere es conocer el valor de IV.

La forma en la que se aplica este tipo de ataque requiere de varias suposiciones por lo que aplicar este ataque en la vida real es difícil, sin embargo se debe de tomar en cuenta cuando se diseña un algoritmo de cifrado. Una posible forma de aplicar satisfactoriamente este ataque se muestra a continuación.

Supongamos que se cifra una imagen R_1 de NxM pixeles bajo una llave k_1 y se obtiene una imagen cifrada T_1, por otra parte tenemos otra imagen R_2 de las mismas dimensiones NxM cifrada con la misma llave k_1. A pesar de que $R_1 \neq R_2$ han sido cifradas con la misma secuencia pseudo-aleatoria, de tal forma que si podemos cifrar una imagen R_3 de las mismas dimensiones en donde todos los pixeles de la imagen son cero, podemos obtener la secuencia como se mostró anteriormente y de esta forma recuperar las imágenes R_1 y R_2.

4.4. Función de cifrado propuesta

Esta función de cifrado se basa nuevamente en el cifrado de Vernam, sin embargo lo que se busca en esta propuesta es que el proceso de cifrado sea aleatorio, para esto debemos incluir elementos estocásticos a la función cifrado, mientras que el proceso de descifrado se mantenga determinista, como se mostró en la sección anterior si añadimos un retardo a la función podemos difundir cambios muy pequeños por toda la imagen.

Para lograr esto, antes de cifrar la imagen realizamos un pre-procesado de la imagen en el cual agregamos una columna de pixeles con valores aleatorios $R(i,1) \in 0,\ldots,255$ para $i = 1,\ldots,N$ con esto el tamaño de la imagen aumentará a Nx$(M+1)$.

Una vez que hemos realizado el pre-procesado ciframos la imagen con la función de retardo de la sección anterior

$$
\begin{aligned}
C_1 &= e(P_1) = P_1 \oplus s_1 \oplus IV; \\
C_i &= e(P_i) = P_i \oplus s_i \oplus C_{i-1}; \\
\\
P_1 &= d(C_1) = C_1 \oplus s_1 \oplus IV; \\
P_i &= d(C_i) = C_i \oplus s_i \oplus C_{i-1};
\end{aligned}
\tag{4.15}
$$

donde P_i es el i-ésimo pixel de la imagen original, cabe mencionar que el pixel $P_1 = R(1,1)$, por lo tanto su tono en escala de grises es aleatorio, C_i es el i-ésimo pixel de la imagen cifrada y s_i es la secuencia pseudo-aleatoria. En este caso usaremos el generador propuesto en la sección 3.2 el cual está basado en mapeos multi-modales, se crearán diferentes secuencias pseudo-aleatorias usando diferentes valores de k. En la figura 4.9 se muestran diferentes imágenes con diferente resolución.

Los resultados de las pruebas de seguridad se discuten a continuación, la primera prueba estadística que aplicamos es la correlación de pixeles adyacentes, en esta ocasión se muestran los resultados de diferentes imágenes originales P y diferentes imágenes cifradas C por medio de diferentes mapeos (uni-modal, bi-modal, tri-modal, cuatri-modal). Los resultados se muestran en la tabla 4.7.

Se puede observar que las diferentes imágenes P presentan una alta correlación de pixeles, mientras que las imágenes cifradas C tienen una baja correlación sin importar el número de modas que se utilizó para crear la secuencia pseudo-aleatoria.

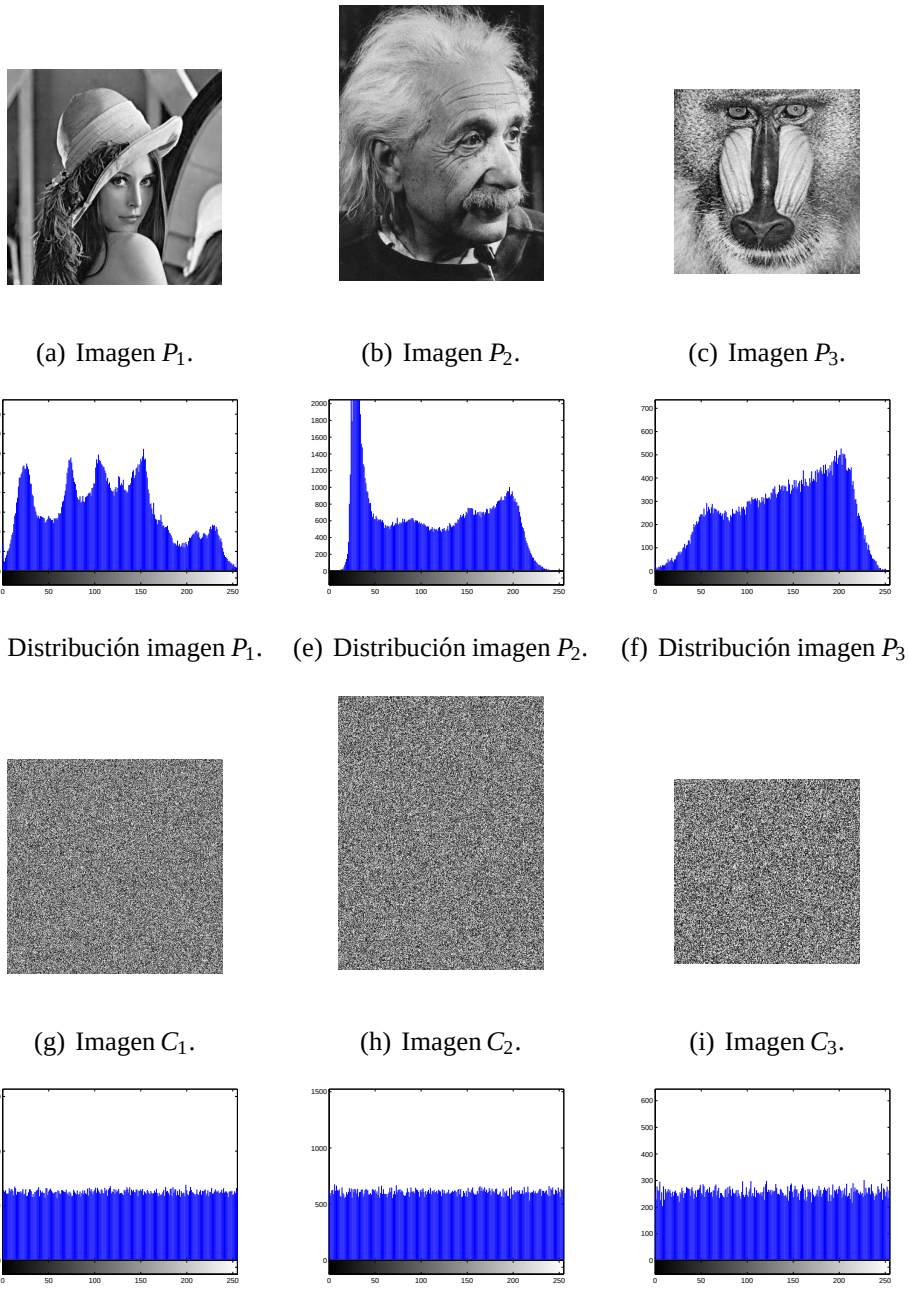
La siguiente prueba estadística evaluada fue la entropía, los resultados de las diferentes imágenes se muestran en la tabla 4.8.

Se puede observar que en algunos casos el valor de la entropía se incrementa ligeramente al incrementar el número de modas mientras que en el peor de los casos la entropía se mantiene al incrementar el número de modas pero en ningún caso se presenta alguna disminución.

La calidad de cifrado la podemos calcular aplicando la ecuación (4.6), se requiere de la imagen original P y la imagen cifrada C para poder evaluar dicha prueba, en la tabla 4.9 se muestran los resultados obtenidos.

Los resultados nos muestran que la calidad de cifrado se incrementa al incrementar el número de modas con el que se crea la secuencia pseudo-aleatoria, esto nos indica que al aumentar el número k se presenta una mayor cantidad de cambios en los valores de los pixeles debido principalmente a que se incrementa la complejidad de secuencia generada.

Los valores de NPCR y UACI se muestran en la tabla 4.10, mantienen los mismos



(a) Imagen P_1. (b) Imagen P_2. (c) Imagen P_3.

(d) Distribución imagen P_1. (e) Distribución imagen P_2. (f) Distribución imagen P_3.

(g) Imagen C_1. (h) Imagen C_2. (i) Imagen C_3.

(j) Distribución imagen C_1. (k) Distribución imagen C_2. (l) Distribución imagen C_3.

Figura 4.9: Cifrado de distintas imágenes.

Tabla 4.7: Correlación de pixeles adyacentes en las imágenes P y C en distintas direcciones.

Imagen	Imagen original	$k=1$	$k=2$	$k=3$	$k=4$
Lenna					
Vertical	0.9852	-0.0280	-0.0051	0.0226	-0.0050
Horizontal	0.9658	0.0252	-0.0113	0.0022	-0.0041
Diagonal	0.9506	0.0176	-0.0157	-0.0081	-0.0077
Einstein					
Vertical	0.9817	0.0091	-0.0245	-0.0164	-0.0001
Horizontal	0.9797	0.0165	-0.0095	0.0085	0.0050
Diagonal	0.9647	0.0136	0.0124	0.0038	0.0035
Mandril					
Vertical	0.9852	-0.0280	-0.0051	0.0226	-0.0050
Horizontal	0.9658	0.0252	-0.0113	0.0022	-0.0041
Diagonal	0.9506	0.0176	-0.0157	-0.0081	-0.0077

Tabla 4.8: Entropía de las imágenes P y C cifradas con diferente número de moda.

Entropía	Imagen original	$k=1$	$k=2$	$k=3$	$k=4$
Lenna	7.8059	7.9988	7.9988	7.9989	7.9989
Einstein	7.7091	7.9970	7.9970	7.9971	7.9973
Mandril	7.4913	7.9989	7.9989	7.9989	7.9989

Tabla 4.9: Calidad de cifrado para imágenes cifradas con $k = 1,2,3,4$.

Calidad de Cifrado	$k=1$	$k=2$	$k=3$	$k=4$
Lenna	267.6016	268.2734	268.6094	269.1953
Einstein	309.7734	311.2969	312.2500	312.6406
Mandril	116.2344	117.5859	117.9062	128.9297

Tabla 4.10: NPCR y UACI de una imagen cifrada dos veces.

	Esquema propuesto
NPCR	99.6094 %
UACI	33.3400 %

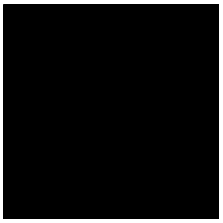
valores que en el cifrado presentado en la sección anterior, esto es debido a que la función de cifrado incluye un retardo y recordemos que con esto garantizamos la difusión de pequeños cambios por toda la imagen. Recordemos que para calcular estos coeficientes se requiere de dos imágenes P_1, P_2 que sean iguales excepto en un pixel y que además sean cifradas con la misma llave k_1 para obtener dos imágenes cifradas C_2, C_2. Con el preprocesado estocástico que se agregó a la función de cifrado nos da la posibilidad de cifrar la imagen P y obtener una imagen C diferente cada que se cifre la imagen incluso con la misma llave k_1, dicho de otra forma el proceso de cifrado se ha convertido en un proceso estocástico en el que cada vez que se aplica el cifrado obtenemos una posible realización de la imagen cifrada de esta forma a partir de P_1 sin realizar ninguna modificación obtendremos $C_1, C_2, C_3, \ldots, C_n$ en cada ocasión que se cifre la imagen. Por lo tanto los valores que se muestran en la tabla 4.10 se obtuvieron a partir de $P_1 = P_2$.

Para ejemplificar las posibilidades de esta función de cifrado mostraremos dos casos, para el primer caso tomamos la imagen de Lenna y aplicamos el proceso de cifrado dos veces con el fin de comparar las imágenes cifradas, para las funciones de cifrado que se mostraron en la sección 4.2 y 4.3 las imágenes cifradas son exactamente iguales, sin embargo al utilizar esta función las imágenes cifradas son diferentes, esto se muestra en la figura 4.10

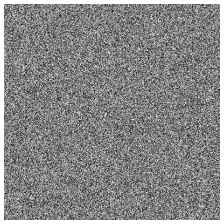
Para el segundo caso se retoma el ejemplo de cifrar una imagen P donde todos los

(a) Imagen P_1.

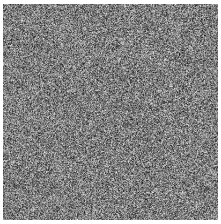
(b) Imagen P_2.



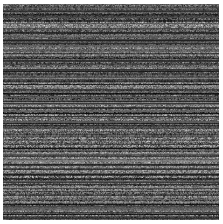
(c) Diferencia de las imágenes P_1, P_2.



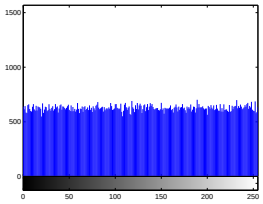
(d) Imagen C_1.



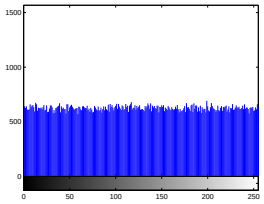
(e) Imagen C_2.



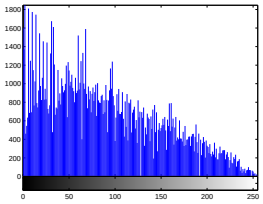
(f) Diferencia de las imágenes C_1, C_2.



(g) Distribución C_1.

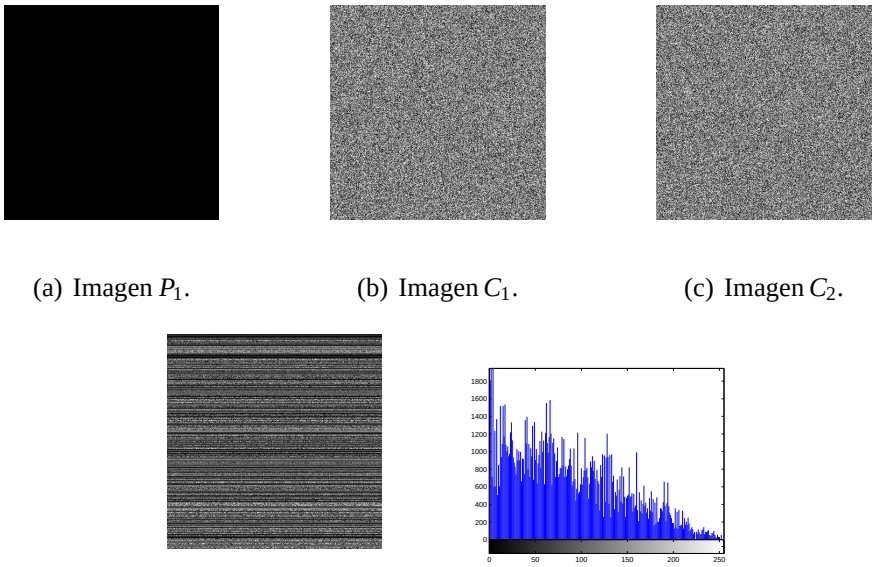


(h) Distribución C_2.



(i) Distribución de la diferencia de las imágenes C_1, C_2.

Figura 4.10: Cifrado de dos imágenes que son idénticas.



(a) Imagen P_1. (b) Imagen C_1. (c) Imagen C_2.

(d) Diferencia de las imágenes C_1, C_2. (e) Distribución de la diferencia de las imágenes C_1, C_2.

Figura 4.11: Cifrado de imagen P en donde todos sus valores son cero.

valores de pixeles son cero, en el cifrado de la sección 4.3 la imagen cifrada C corresponde a la secuencia pseudo-aleatoria generada, en este caso la imagen cifrada corresponde a una posible realización de la imagen cifrada y además si ciframos nuevamente la imagen P bajo la misma llave k_1 obtendremos una imagen $C_2 \neq C_1$, en la figura 4.11 se muestran las imágenes P, C_1, C_2 y la diferencia entre las imágenes C_1, C_2, con esto la secuencia generada no queda expuesta de forma directa , sin embargo, se puede modificar la función de cifrado como se muestra a continuación:

$$\begin{aligned} C_1 &= e(P_1) = P_1 \oplus s_1 \oplus IV; \\ C_i &= e(P_i) = P_i \oplus s_i \oplus C_{i-1} \oplus C_{i-1} = P_i \oplus s_i. \end{aligned} \tag{4.16}$$

de tal forma que si tomamos el caso en donde todos los píxeles de P son cero, podemos reconstruir la secuencia pseudo-aleatoria s_i, ya que básicamente la función de cifrado se transforma en la propuesta por Vernam, con la ventaja de que la función es aleatoria y no determinista, pero esta propiedad no implica la resistencia de ataques por texto escogido.

En este trabajo de tesis nos enfocamos principalmente en los sistemas caóticos de tiempo discreto, sin embargo, realizamos una extensión de este trabajo y lo aplicamos a sistemas caóticos disipativos inestables, estos sistemas son capaces de formar multi-

enroscados por medio de una ley de conmutación, en base a esto se propuso un generador de bits pseudo-aleatorio el cual muestra resultados satisfactorios a las pruebas de aleatoriedad del NIST y posteriormente se utilizaron para cifrar imágenes en escala de grises, esta información se encuentra en el apéndice C.

Capítulo 5

Conclusiones

Este trabajo de tesis doctoral fue presentado en tres partes. En la primera parte se presentó el estudio de los mapeos caóticos haciendo énfasis en los mapeos multi-modales, con esto se obtuvieron dinámicas útiles para cifrados en flujo. En la segunda parte se presentaron dos metodologías para crear generadores pseudo-aleatorios, además de presentar herramientas de los sistemas dinámicos que son de utilidad para definir los parámetros de dichos sistemas, así como las herramientas de la criptografía que en este caso son las pruebas estadísticas para evaluar la aleatoriedad de las secuencias generadas. Por último en la tercera parte se mostró la aplicación de los conceptos en imágenes, además de evaluar las funciones de cifrado contra ataques de tipo diferencial y de texto escogido. Con todo lo anterior este trabajo de tesis nos permite establecer las siguientes conclusiones.

- Es posible obtener un mapeo bi-modal a partir de los mapeos logístico y casa de campaña los cuales son uni-modales. Se mostró que efectivamente este mapeo bi-modal presenta comportamiento caótico, esto fue demostrado por medio del exponente de Lyapunov, así como por medio de satisfacer la definición de Devaney. Para mapeos multi-modales se mostró el comportamiento caótico por medio del exponente de Lyapunov, sin embargo es posible extender la demostración por medio de la definición de Devaney del mapeo bi-modal para cualquier número de modas. Por otro lado se realizó la implementación electrónica del mapeo bi-modal por medio de componentes analógicos, cabe señalar que es la primera implementación experimental de mapeos con más de una moda el cual puede ser utilizado para la generación de secuencias aleatorias.

- Se mostró que el mapeo logístico puede tomar valores negativos en el parámetro

de bifurcación, con esto se amplía el rango de valores válidos para el parámetro α, se pudo observar que se mapea a valores diferentes con α negativo, sin embargo el exponente de Lyapunov presenta simetría. También se mostró que por medio del uso de retardos es posible eliminar la forma del mapeo en el espacio fase por lo que reconstruir el mapeo a partir de una serie de tiempo resultaría demasiado complejo. Además se mostró que la combinación de dos series de tiempo con retardo y parámetro de bifurcación positivo y negativo, dan lugar a secuencias pseudo-aleatorias que son seguras para su uso en el campo de la criptografía, se puede hacer esta aseveración ya que el generador fue evaluado con las pruebas estadísticas de aleatoriedad propuestas por el NIST las cuales son un estándar internacional en el campo de la criptografía.

- Se propuso un generador pseudo-aleatorio basado en mapeos multi-modales, cabe señalar que es el primer generador que utiliza este tipo de mapeos, entre las ventajas que presentan estos mapeos es que pueden cambiar el número de modas al variar solamente el parámetro de bifurcación y además presentan un amplio rango de valores para el parámetro de bifurcación. De forma similar al generador anterior se propuso la mezcla de series de tiempo con el fin de eliminar la forma del mapeo en el espacio fase, sin embargo el uso de estos mapeos nos permite incrementar el número de series mezcladas sin tener que definir nuevas funciones para cada mapeo. Finalmente se evaluó el generador propuesto por medio de las pruebas de aleatoriedad del NIST y se mostró que a partir de la combinación de dos secuencias, se obtienen series binarias que son seguras para su uso en la criptografía.

- Se implementaron criptosistemas de cifrado en flujo usando los generadores propuestos, para evaluar a los cifrados se requiere de pruebas estadísticas, estas pruebas se les conoce como pruebas de seguridad y son diferentes a las pruebas de aleatoriedad. Con esto se mostró que aunque la secuencia generada sea criptográficamente segura, el cifrado puede resultar inseguro si la función de cifrado no es diseñada de forma correcta, además de la evaluación de las pruebas de seguridad se debe someter a la función de cifrado a ataques por lo menos los más comunes, como se observó en este trabajo de tesis las pruebas de seguridad son necesarias pero no suficientes. Se mostró y se evaluó la resistencia ataques de tipo diferencial, en donde para lograr evitar este ataque se añadió un retardo a la función de cifrado y de esta forma se pueden difundir pequeños cambios por toda la imagen.

Por otro lado se propone un pre-procesado a la función de cifrado de tal forma que presente elementos aleatorios, con esto al cifrar la imagen deja de ser un proceso determinista y se obtiene una posible realización de la imagen cifrada, sin embargo del proceso de descifrado es determinista por lo que sin importar de que posible realización provenga la imagen cifrada siempre es posible descifrar la imagen.

Con el desarrollo de este trabajo de tesis nos permite visualizar como trabajo a futuro algunos aspectos que se mencionan brevemente a continuación:

- Por un lado se tiene el interés de seguir trabajando en los cifrados en flujo, la idea es proponer nuevas metodologías para la obtención de secuencias binarias a partir de las series de tiempo que generan los mapeos. En este caso la metodología propuesta está relacionada con la división del espacio fase, sin embargo, existe una gran cantidad de variantes que son capaces de producir secuencias binarias que pasen las pruebas estadísticas del NIST.

- Por otro lado el siguiente paso es trabajar con cifrados en bloque, en primera instancia se trabajará con una metodología para obtener cajas de sustitución basadas en mapeos caóticos, así como el desarrollo de pruebas estadísticas que nos permitan establecer una medida de la calidad de la caja de sustitución.

- Por último se tiene el desarrollo de esquemas de cifrado que incluyan un módulo extra de tal forma que nos permita garantizar tanto la confidencialidad como la integridad, todo lo anterior basado en sistemas caóticos.

Apéndice A

Implementación electrónica del mapeo bi-modal

La generación de números aleatorios puede ser dada por medio de la implementación electrónica de un sistema dinámico discreto el cual consta de dos partes: (1) el circuito del mapeo y (2) el circuito que se encarga del proceso iterativo, en la figura A.1 se muestra un diagrama general del circuito. Algunas implementaciones electrónicas de sistemas dinámicos discretos están basadas en microprocesadores o microcontroladores, ya sea una o ambas partes del sistema [175], sin embargo esto nos lleva a discretizar el espacio X. De acuerdo a la definición de Devaney (definición 10), podemos ver que si discretizamos el espacio X no se cumple la definición de caos. Por lo tanto un sistema caótico debe de implementarse por medio de dispositivos analógicos. Los sistemas dinámicos discretos con comportamiento caótico al ser implementados electrónicamente pueden generar secuencias aleatorias debido al ruido presente en todo dispositivo electrónico. Por otro lado, un generador de números pseudo-aleatorios (PRNG) utiliza una o más entradas y genera múltiples números "pseudo-aleatorios". Las entradas a un PRNGs se llaman semillas. En contextos en los que hace falta la imprevisibilidad, la semilla misma debe ser aleatoria e impredecible. De ahí que, por defecto, un PRNG debería obtener sus semillas a partir de las salidas de un generador de números aleatorios; es decir, un PRNG requiere un TRNG como compañero.

Para esto nos enfocaremos primero en el desarrollo del circuito del mapeo por medio de amplificadores operacionales, multiplicadores analógicos, diodos y resistencias. En la figura A.2 se muestra un diagrama a bloques del mapeo bi-modal. La salida tiene tres gran-

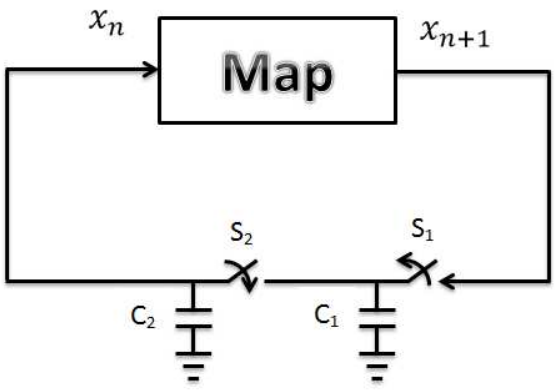


Figura A.1: Diagrama general de un mapeo electrónico.

des ramas: la primera genera el mapeo logístico (nodo A), las otras 2 ramas corresponden al mapeo casa de campaña (nodo B y C). Típicamente, los circuitos electrónicos están formados por varios amplificadores operacionales, los cuales realizan operaciones lineales (por ejemplo: integración y sumatorias) así como un conjunto de circuitos integrados que realizan operaciones no lineales (multiplicación), este circuito combina componentes activos y pasivos y es capaz de reproducir la transición del estado estable al caos, como se observa en el diagrama de bifurcación (figura 2.18).

La figura A.3 muestra el diagrama esquemático del circuito mapeo bi-modal, la salida del circuito será analizada usando los voltajes de los nodos: A, B, C, D.

El voltaje en el nodo A esta dado por el multiplicador M1 el cual tiene cuatro entradas (x_1, x_2, y_1, y_2) y una salida la cual entrega el voltaje $W = \frac{(x_1-x_2)(y_1-y_2)}{10}$. El uso de multiplicadores analógicos requiere emplear una normalización por un factor de 10, esto se debe a restricciones físicas que imponen los multiplicadores. Las entradas $x_1 = V_{in}(R_2R_4)/(R_1R_3)$ y $y_2 = V_{in}(R_2R_6)/(R_1R_5)$ están dadas por los amplificadores operacionales U2 y U3, respectivamente. Por lo tanto el voltaje en el nodo A es el siguiente:

$$V_A = \left(V_{in}\frac{R_2R_4}{R_1R_3}\right)\left(5 - V_{in}\frac{R_2R_6}{R_1R_5}\right)/10. \tag{A.1}$$

Después de sustituir los componentes con los valores de la tabla A.1, el voltaje en el nodo A es $V_{in}(1-V_{in})$, esta señal corresponde al mapeo logístico sin considerar el parámetro de bifurcación α.

El voltaje en el nodo B está dado por la salida del amplificador operacional U4 el cual

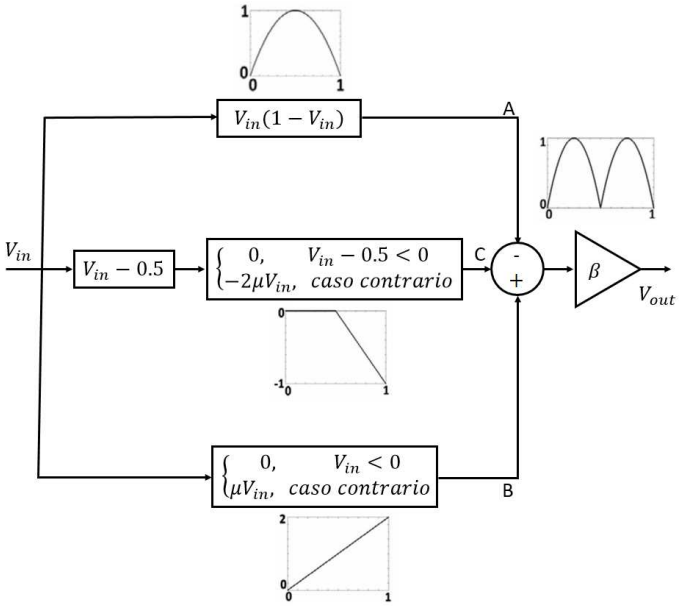


Figura A.2: Diagrama a bloques del mapeo bi-modal usado para la construcción de su implementación electrónica.

tiene una retroalimentación en la entrada negativa, el voltaje de salida es:

$$V_B = -V_{in}R_{13}/R_{12}. \tag{A.2}$$

El voltaje en el nodo C está dado por la salida del amplificador operacional U5 la cual es una señal por partes como se muestra:

$$V_C \;=\; \begin{cases} 0, & \text{para } V_{in} < \frac{R_7}{2R_8}; \\ \frac{R_{11}}{R_{10}}\left(\frac{R_9 V_{in}}{R_7} - \frac{R_9}{2R_8}\right), & \text{para } V_{in} \geq \frac{R_7}{2R_8}. \end{cases} \tag{A.3}$$

Las ecuaciones (A.1) y (A.2) corresponden al mapeo casa de campaña, recordemos que $f_T(x,\mu)$ está definido por dos partes, para asegurar que el mapeo casa de campaña es simétrico el parámetro de bifurcación μ debe ser el mismo en ambos lados. Podemos observar que μ está dado por R_{13}/R_{12} y $R_{11}/(2R_{10})$, esto nos lleva a las siguientes restricciones: $R_{11} = 2R_{13}$ y $R_{10} = R_{12}$.

La salida del amplificador U7 es la suma del voltaje de los nodos A, B y C, esta suma corresponde al voltaje en el nodo D y está dado por la siguiente ecuación:

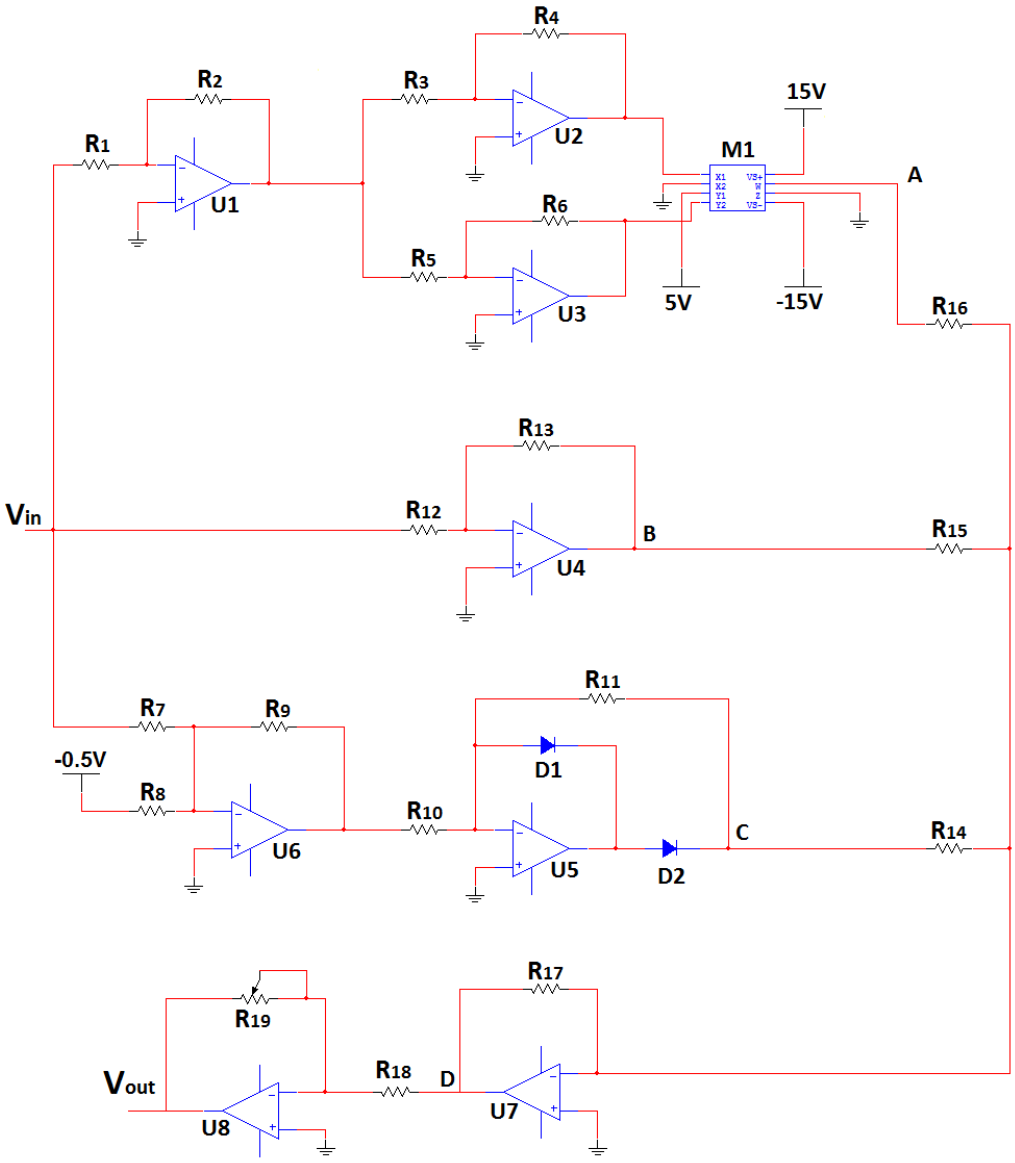


Figura A.3: Diagrama esquemático del circuito del mapeo bi-modal.

Tabla A.1: Valores de los componentes electrónicos empleados en la construccion del circuito del mapeo bi-modal.

Dispositivo	Valor
$R_1, R_3, R_5, R_6, R_7, R_8, R_9,$	
$R_{10}, R_{12}, R_{16}, R_{18}$	Resistencia de 10kΩ
R_4	Resistencia de 4kΩ
$R_{11}, R_{14}, R_{15}, R_{17}$	Resistencia de 40kΩ
R_{13}	Resistencia de 20kΩ
R_{19}	Potenciómetro de 40kΩ
D1,D2	Diodo 1N4148
U1,U2,U3,U4,U5,U6,U7,U8	Amplificador operacional TL084
M1	Multiplicador AD633

$$V_D = -R_{17}\left(\frac{V_A}{R_{16}} + \frac{V_B}{R_{15}} + \frac{V_C}{R_{14}}\right). \tag{A.4}$$

Cabe mencionar que la proporción R_{17}/R_{16} es el parámetro $\alpha = 4$. Por lo tanto el voltaje en el nodo D es $(-f_L + f_T)$ que de hecho es el mapeo bi-modal invertido y sin tomar en cuenta el parámetro de bifurcación β.

Por último, el voltaje V_{out} está dado por el amplificador inversor U8, la salida es $(R_{19}/R_{18})V_D$. Asumiendo que todos los componentes se comportan de forma ideal, la salida del circuito presentado en la figura A.3 es modelado por la siguiente ecuación:

$$V_{out} = \frac{R_{19}}{R_{18}} \begin{cases} 4V_{in}(1-V_{in}) - 2V_{in}, & \text{para } V_{in} < \frac{1}{2}V; \\ 4V_{in}(1-V_{in}) + 2V_{in} - 2, & \text{para } V_{in} \geq \frac{1}{2}V. \end{cases} \tag{A.5}$$

La ecuación (2.16) se puede obtener de la ecuación (A.5) con cambios de variables $V_{in} = x_n, V_{out} = x_{n+1}$ y $\beta = R_{19}/R_{18}$.

La segunda parte del circuito se encarga del proceso iterativo (ver figura A.1), este circuito consiste de un microcontrolador PIC16F88 de Microchip y dos hold and sample LF398 de National Semiconductors, el fin es mantener el voltaje de la señal V_{out} dado por la ecuación (A.5). Dicho de otra forma los circuitos hold and sample son usados como memorias analógicas para almacenar el valor de x_n y obtener x_{n+1}, de esta forma realizamos el proceso iterativo del mapeo. La figura A.4 muestra el diagrama esquemático

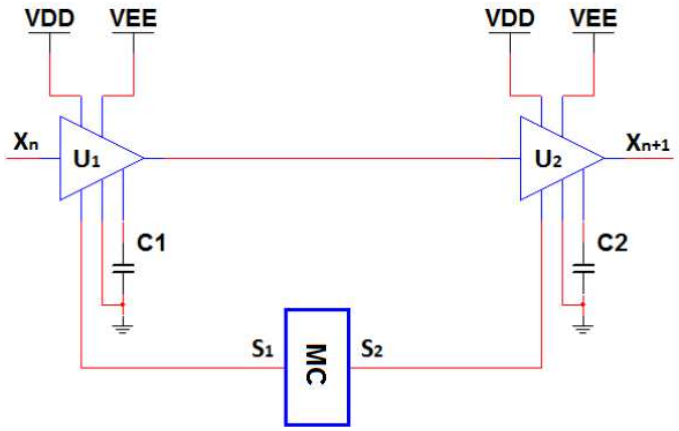


Figura A.4: Diagrama esquemático del circuito iterativo, U1 y U2 son los dispositivos LF398 y MC es el microcontrolador PIC16F88.

de esta parte del circuito, se puede observar que los dispositivos LF398 (U1 y U2) tienen una entrada de activación, la señal para ambos dispositivos hold and sample proviene del microcontrolador PIC16F88.

El tiempo de disparo para activar cada dispositivo es definido por el diseñador, en este caso existe un tiempo de 20ms entre cada disparo y la duración del disparo es de 1ms, estos tiempos son programados en el microcontrolador y pueden variar dependiendo de la aplicación. La figura A.5 muestra un diagrama con los tiempos de activación de cada uno de los dispositivos hold and sample.

Una vez que los dos circuitos han sido configurados correctamente el mapeo bi-modal comienza el proceso iterativo. La figura A.6 muestra la serie de tiempo experimental del mapeo con $\beta = 4$. A pesar de la reactancia parásita, el ancho de banda finito de los componentes activos y otras perturbaciones experimentales como el ruido, el circuito electrónico muestra un comportamiento cercano al modelo matemático dado por la ecuación (2.16). Para esta realización experimental los amplificadores operacionales TL084 y los dispositivos hold and sample LF398 fueron alimentados con una fuente de voltaje de ± 15 V, además se requiere de una fuente de voltaje adicional de -0.5 V para el correcto funcionamiento de mapeo casa de campaña.

El valor del parámetro de bifurcación β puede ser fijado ajustando el potenciómetro R_{19} localizado en el amplificador operacional U8. Para explorar el rango completo de la dinámica que presenta este circuito, se han ajustado diferentes valores de R_{19}, este potenciómetro puede variar en el intervalo $[0\Omega, 40k\Omega]$. Se realizó un barrido con los valores

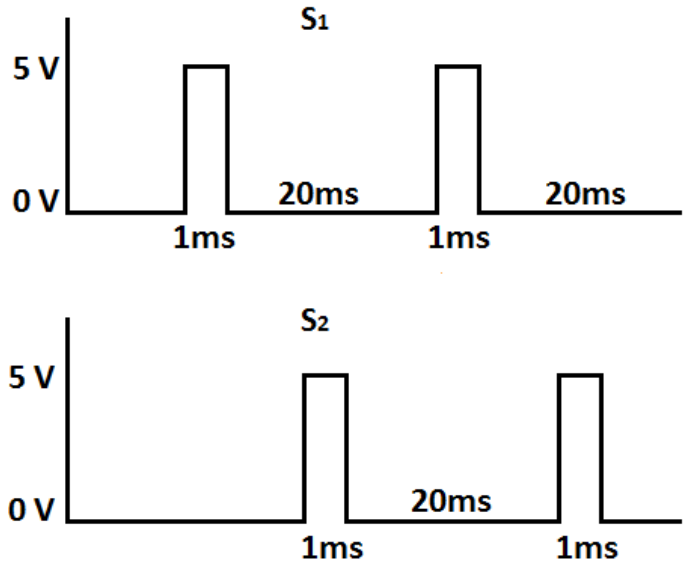


Figura A.5: Tiempos de activacion para los dispositivos hold and sample.

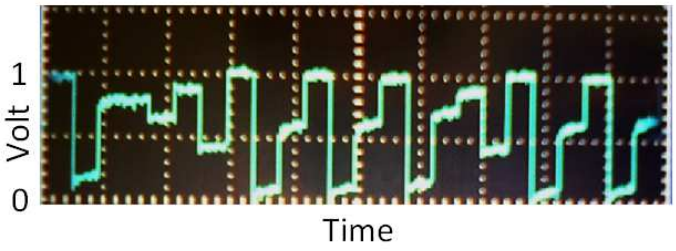


Figura A.6: Serie de tiempo con dinámica caótica generada por el mapeo bi-modal con $\beta = 4$.

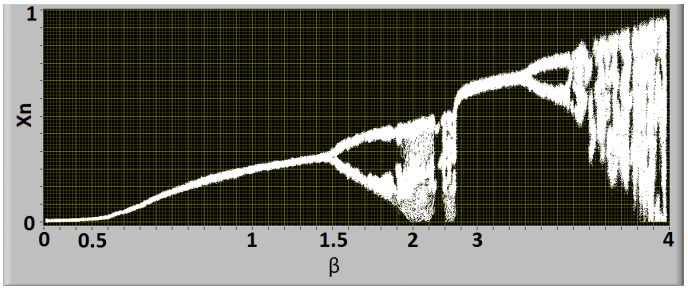


Figura A.7: Diagrama de bifurcación experimental del mapeo bi-modal.

válidos de β para obtener el diagrama de bifurcación experimental el cual se muestra en la figura A.7, donde se pueden observar: puntos fijos, oscilaciones periódicas, cascadas de periodo dos, bifurcaciones y caos, por lo que el circuito muestra los diferentes comportamientos del mapeo bi-modal, además se tiene buena concordancia con las simulaciones numéricas.

Apéndice B

Pruebas estadísticas

El análisis estadístico de las secuencias generadas tiene como objetivo comprobar si cada una de las secuencias posee características que nos permita suponer su aparente aleatoriedad. Actualmente no existe alguna prueba matemática que nos permita afirmar de forma analítica la propiedad de la aleatoriedad por esta razón utilizamos un conjunto de pruebas estadísticas en donde la conclusión de cada prueba es independiente de las demás, de tal forma que si dichas secuencias superan las pruebas podemos aceptar la secuencia como aleatoria, sin embargo no es una afirmación definitiva.

En la sección 2.1.1 se mostró una variedad de generadores, en donde el objetivo principal de esta tesis son los generadores pseudo-aleatorios criptográficamente seguros, en los que se pide la propiedad especial de impredecibilidad, al tener dicha propiedad se pueden realizar las siguientes suposiciones al ser evaluadas por las pruebas estadísticas:

- Uniformidad: En cualquier punto de la generación de una secuencia aleatoria o pseudo-aleatoria, la ocurrencia de un cero o un uno es igualmente probable, por lo tanto la probabilidad de cada uno es exactamente $1/2$. El número esperado de ceros o unos es $n/2$, donde n es la longitud de la secuencia.

- Escalabilidad: Cualquier prueba aplicable a una secuencia, podrá ser aplicada también a subsecuencias extraídas aleatoriamente. Si la secuencia es aleatoria, entonces cada subsecuencia extraída también será aleatoria, por lo tanto cualquier subsecuencia extraída deberá pasar las pruebas estadísticas.

- Consistencia: El comportamiento de un generador será consistente con el valor inicial, es decir con la llave. Dicho de otra forma si la llave es un parámetro débil

entonces la secuencia generada también lo será, por lo tanto es inadecuado comprobar un generador pseudo-aleatorio basado en la utilización de un valor inicial débil o a partir de una salida física poco aleatoria.

El banco de pruebas estadísticas del NIST [173] consiste en 15 pruebas que fueron desarrolladas para probar la aleatoriedad de secuencias binarias de longitud arbitraria, estas secuencias pueden ser producidas por hardware o por software. Estas pruebas se enfocan en diferentes puntos para tratar de encontrar no-aleatoriedad que pudiera existir en una secuencia. A continuación se muestra una breve descripción de cada una de las pruebas:

- Prueba de frecuencia (Mono-bit): Esta prueba verifica si el número de unos y ceros en una secuencia son aproximadamente iguales como se esperaría en una secuencia realmente aleatoria.

- Prueba de frecuencia dentro de un bloque (128 bits): Esta prueba determina si la frecuencia de unos en un bloque de M=128 bits es aproximadamente M/2. Para un bloque con M=1, esta prueba se convierte a la prueba de frecuencia de 1 bit (Mono-bit).

- Prueba de corridas: Una corrida de longitud *k* consiste en exactamente *k* bits idénticos y están limitados antes y después con un bit de valor opuesto. Esta prueba determina si el número de corridas de unos y ceros de varias longitudes coincide con los valores que se esperarían en una secuencia aleatoria. En particular con esta prueba se determina si la oscilación entre ceros y unos es rápida o lenta.

- Prueba de corrida larga: Esta prueba determina si la longitud de la corrida más larga de unos dentro de la secuencia evaluada es consistente con la longitud de la corrida más larga de unos que se esperaría en una secuencia aleatoria. Si se encuentra alguna irregularidad en la longitud de la corrida mas larga de unos, esto implica que también se encontrará alguna irregularidad en la corrida más larga de ceros, por esta razón solo es necesario realizar la prueba sobre un valor.

- Prueba de rango de una matriz binaria: Esta prueba verifica la dependencia lineal entre subsecuencias de longitud fija de la secuencia original, para esta prueba es necesario construir matrices con 1024 elementos (32×32).

- Prueba de transformada discreta de Fourier: Esta prueba detecta características periódicas en la secuencia evaluada, con esta prueba se encuentran patrones que están cercanos unos de otros.

- Prueba de búsqueda de patrones no sobrepuestos: Esta prueba y la prueba de búsqueda de patrones sobrepuestos usan una ventana de M=9 bits en busca de patrones específicos, con el propósito de detectar generadores que produzcan demasiadas ocurrencias de un patrón no periódico. Si el patrón no es encontrado, la ventana se desplaza un bit. Si el patrón es encontrado la ventana se ubica un bit después del patrón encontrado y la búsqueda continua.

- Prueba de búsqueda de patrones sobrepuestos: La diferencia entre esta prueba y la prueba de no-sobrepuestos es que cuando un patrón es encontrado, la ventana se desplaza solo un bit y la búsqueda continúa.

- Prueba estadística universal de Maurer: Esta prueba detecta si la secuencia puede ser significativamente comprimida sin pérdida de información. Si la secuencia puede ser comprimida entonces se considera no aleatoria.

- Prueba de complejidad lineal: Esta prueba determina si una secuencia es lo suficientemente compleja para ser considerada como aleatoria, las secuencias aleatorias son examinadas por medio de registros de corrimiento de retroalimentación lineal (LSFR). Si la secuencias puede ser reproducida por un LSFR corto entonces implica que la secuencia no es aleatoria.

- Prueba serial: Esta prueba calcula la frecuencia de todos los posibles patrones de superposición de m-bits a través de toda la secuencia. Así determina si el número de ocurrencias de los patrones superpuestos 2^M m-bits es aproximadamente el mismo que el que se esperaría para una secuencia aleatoria. Una secuencia aleatoria tiene una distribución uniforme, esto es cada subsecuencia tiene la misma probabilidad de aparecer, si M=1 es equivalente a la prueba de frecuencia, en nuestro caso utilizamos M=16.

- Prueba de entropía aproximada: Esta prueba compara la frecuencia de traslape de dos bloques consecutivos, en esta prueba se tomaron bloques de tamaño M=10.

- Prueba de sumas acumulativas: Esta prueba determina si la suma acumulativa en la secuencia evaluada (-1,+1) es como se esperaría para una secuencia aleatoria. Para

calcular esta prueba el NIST ha propuesto dos algoritmos diferentes, en sentido hacia delante y en sentido hacia atrás, en ambos casos la suma acumulativa debe ser cercana a cero.

- Prueba de excursiones (caminatas) aleatorias: Esta prueba determina si el número de visitas a un estado en particular dentro de un ciclo coincide con el que se esperaría en una secuencia aleatoria. Esta prueba contiene 8 subpruebas y una conclusión para cada uno de los estados: -4, -3, -2, -1, 1, 2, 3, 4.

- Prueba de excursiones (caminatas) aleatorias variantes: Esta prueba detecta variaciones del número esperado de visitas a varios estados en una caminata aleatoria. Esta prueba contiene 18 subpruebas y una conclusión para cada uno de los estados:$-9,-8,-7,-6,-5,-4,-3,-2,-1,1,2,3,4,5,6,7,8,9$.

Para la interpretación de resultados existen dos formas 1) por medio de la porción de secuencias que pasan las pruebas y 2) por medio de la distribución de los valores "P-value". En este trabajo de tesis tomamos el primer criterio para la evaluación de resultados. Para esto es necesario la definición de un valor al que se le conoce como nivel de significancia y se denota como σ, típicamente este valor se define entre el rango [0.001,0.01]. Cuando $\sigma = 0.001$, se espera que las pruebas estadísticas rechacen una secuencia entre mil, para este trabajo definimos el valor $\sigma = 0.01$ con lo cual se espera que las pruebas rechacen una secuencia entre cien.

Una vez que se ha definido el nivel de significancia podemos calcular los límites que definen al intervalo de confianza, si el resultado de una prueba estadística cae dentro de este intervalo, entonces podemos decir que la secuencia tiene las mismas características que una secuencia realmente aleatoria y por lo tanto pasa la prueba estadística. El intervalo de confianza se calcula de la siguiente forma:

$$L_{1,2} = (1-\sigma) \pm 3\sqrt{\frac{1-(1-\sigma)}{m}}, \tag{B.1}$$

donde $\sigma = 0.01$ es el nivel de significancia y m es el número de secuencias que contiene la muestra.

Apéndice C

Sistemas caóticos disipativos inestables

C.1. Sistemas disipativos inestables

Los sistemas lineales por partes en $\mathbf{R}^3$ se presentaron en [176, 177] y se extendieron a $\mathbf{R}^4$ en [178], para esto consideremos un sistema lineal afín dado por

$$\dot{\mathbf{X}} = \mathbf{A}\mathbf{X} + \mathbf{B}, \tag{C.1}$$

donde $\mathbf{X} = [x_1, x_2, x_3, x_4]^T \in \mathbf{R}^4$ es el vector de estados, $\mathbf{A} = [a_{ij}] \in \mathbf{R}^{4\times 4}$, $i, j = 1, 2, 3, 4$, denota una matriz y $\mathbf{B} = [B_1, B_2, B_3, B_4]^T \in \mathbf{R}^4$ representa un vector. Estamos interesados en un sistema disipativo que tenga puntos de equilibrio hiperbólicos en $\mathbf{X}^*$, por ejemplo $\mathbf{A}\mathbf{X}^* + \mathbf{B} = 0$. El correspondiente conjunto de eigenvalores $\bigwedge = \lambda_i, i = 1, \ldots, 4$ de $\mathbf{A}$ son: dos λ_i son reales negativos y dos λ_i son complejos conjugados con parte real positiva. Para asegurar que la ecuación (C.1) es disipativa, los eigenvalores deben de satisfacer $\sum_{i=1}^{4} \lambda_i < 0$.

El sistema dado por la ecuación (C.1) es inestable, por lo tanto, consideremos el siguiente sistema conmutado dado por:

$$\dot{\mathbf{X}} = \mathbf{A}\mathbf{X} + \mathbf{B}(\mathbf{X}),$$

$$\mathbf{B}(\mathbf{X}) = \begin{cases} B_1, & \text{si } \mathbf{X} \in \mathcal{D}_1; \\ B_2, & \text{si } \mathbf{X} \in \mathcal{D}_2; \\ \vdots & \vdots \\ B_k, & \text{si } \mathbf{X} \in \mathcal{D}_k; \end{cases} \tag{C.2}$$

donde $\mathbf{R}^4 = \cup_{i=1}^{k} \mathcal{D}_i$. El sistema dado por la ecuación (C.2) tiene los puntos de equilibrio $\mathbf{X}_1^* \in \mathcal{D}_1, \ldots, \mathbf{X}_k^* \in \mathcal{D}_k$ con $\mathbf{A}\mathbf{X}_i^* + B_i = 0, i = 1, \ldots, k$. El objetivo es elegir vectores B_i de tal forma que el sistema (C.2) presente comportamiento caótico. Para lograr esto, debemos tener una colección de órbitas heteroclínicas $\phi(\mathbf{X}_0)$ atrapadas en un atractor hipercaótico $\mathfrak{A}$, teniendo en cuenta esto se deben definir al menos dos vectores B_1 y B_2 que conecten a los puntos de equilibrio. Es importante mencionar que todas estas conexiones heteroclínicas son estructuralmente estables.

El sistema generado por este método puede mostrar k multi-enroscados como resultado de la combinación de varias trayectorias inestables, donde k es el número de subsistemas.

La ubicación de los enroscados ocurre en el mismo eje en el cual pertenecen los puntos de equilibrio de los subsistemas. Puede existir una gran variedad de sistemas que satisfagan las restricciones mencionadas, sin embargo, en este caso la matriz $\mathbf{A}$ y el vector $\mathbf{B}$ se definieron como se muestra a continuación:

$$\mathbf{A} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -1.5 & -1 & -1 & 0 \\ 0 & -1 & 0 & -1 \end{pmatrix}, \mathbf{B} = \begin{pmatrix} 0 \\ 0 \\ B_3 \\ B_4 \end{pmatrix}. \tag{C.3}$$

En este caso consideramos $B_3 = B_4$, además los puntos de equilibrio se obtuvieron de $\mathbf{X}^* = -\mathbf{A}^{-1}\mathbf{B}$ por lo que se desplazaran en los ejes x_1 y x_4. Si se considera otra matriz $\mathbf{A}$ y un vector diferente $\mathbf{B}$ el desplazamiento puede darse en diferente dirección.

Consideremos la transformación lineal $T : \mathbf{R}^4 \to \mathbf{R}^4$ con la siguiente transformación de coordenadas:

$$\dot{\mathbf{Y}} = \hat{\mathbf{A}}\mathbf{Y} + \hat{\mathbf{B}}. \tag{C.4}$$

donde $\mathbf{Y} = \mathbf{Q}^{-1}\mathbf{X}$, $\mathbf{Q}$ es una matriz invertible que satisface las siguientes condiciones: $\hat{\mathbf{A}} = \mathbf{Q}^{-1}\mathbf{A}\mathbf{Q}$ y $\hat{\mathbf{B}} = \mathbf{Q}^{-1}\mathbf{B}$, tomando los siguientes valores:

$$\hat{\mathbf{A}} = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 0.1020+1.1115i & 0 & 0 \\ 0 & 0 & 0.1020-1.1115i & 0 \\ 0 & 0 & 0 & -1.2041 \end{pmatrix}, \qquad \hat{\mathbf{B}} = B_3 \begin{pmatrix} 3 \\ 0.4223+0.3434i \\ 0.4223-0.3434i \\ 2.1329 \end{pmatrix}. \tag{C.5}$$

Por lo tanto, el conjunto de eigenvalores del sistema dado por la ecuación (C.3) es el siguiente: $\Lambda = diag(\hat{\mathbf{A}}) = \{-1.0000, 0.1020 \pm 1.1115i, -1.2041\}$.

Ahora definiremos una ley de conmutación que dependa del valor de x_1 que resulta en un atractor de dos enroscados, se representa como se muestra a continuación:

$$B_3 = \begin{cases} 0.9, & \text{si } x_1 \geq 0; \\ 0 & \text{de otra forma.} \end{cases} \tag{C.6}$$

El sistema dado por la ecuación (C.3) en conjunto con la ley de conmutación dada por la ecuación (C.6) presenta los siguientes puntos de equilibrio: $\mathbf{X}_1^* = (0,0,0,0)^T$ y $\mathbf{X}_2^*$ =(0.6, 0, 0, 0.9)^T. Dado que $\mathbf{R}^4$ es dividido en dos subsistemas con un punto de equilibrio, para cada uno de estos puntos emerge un enroscado. La proyección de este atractor en el plano (x_1, x_2) con la condición inicial $\mathbf{X}_0 = (1,0,0,0)^T$ presenta doble enroscado (figura C.1 a).

Si la ley de conmutación es diseñada considerando más subsistemas que dividan B_3, entonces se pueden obtener más enroscados en el atractor resultante. Para obtener 4 enroscados, B_3 se puede definir como sigue:

$$B_3 = \begin{cases} 1.8, & \text{si } x_1 \geq 0.9; \\ 0.9, & \text{si } 0.3 \leq x_1 < 0.9; \\ 0, & \text{si } -0.3 < x_1 < 0.3; \\ -0.9, & \text{if } x_1 \leq -0.3. \end{cases} \tag{C.7}$$

Esta ley de conmutación agrega dos subsistemas adicionales con sus correspondientes puntos de equilibrio localizados en $\mathbf{X}_3^* = -\mathbf{X}_2^*$ y $\mathbf{X}_4^* = (1.2,0,0,1.8)^T$. La proyección en

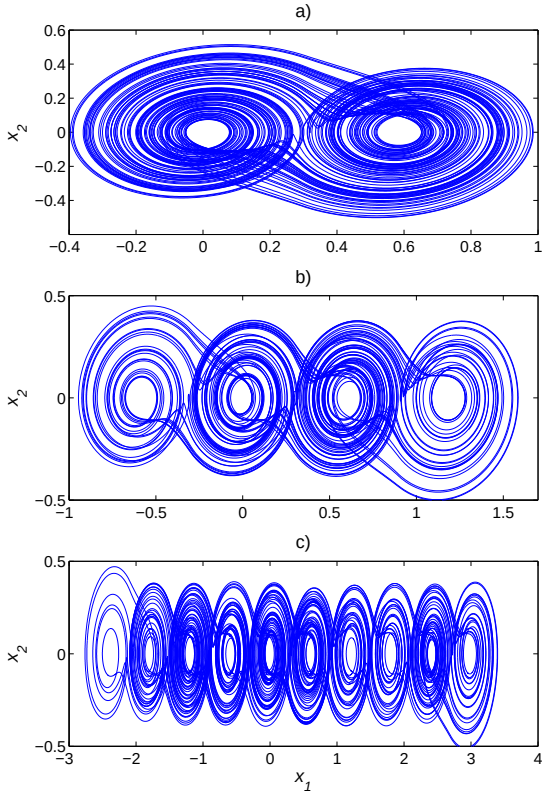


Figura C.1: Proyección del sistema dado por la ecuación (C.2) con (C.3) sobre el plano (x_1, x_2) para diferentes leyes de conmutación: a) 2 enroscados con la ecuación (C.6); b) 4 enroscados con la ecuación (C.7); c) 10 enroscados con la ecuación (C.8).

el plano (x_1, x_2) dada por la ley de conmutación (C.7) con condiciones iniciales $\mathbf{X}_0 = (1,0,0,1)^T$ produce 4 enroscados en el atractor (figura C.1 b).

Por último consideremos agregar 6 subsistemas, con esto tendremos el caso de 10 enroscados en el atractor, para esto se define la siguiente ley de conmutación:

$$B_3 = \begin{cases} 4.5, & \text{si } x_1 \geq 2.7; \\ 3.6, & \text{si } 2.1 \leq x_1 < 2.7; \\ 2.7, & \text{si } 1.5 \leq x_1 < 2.1; \\ 1.8, & \text{si } 0.9 \leq x_1 < 1.5; \\ 0.9, & \text{si } 0.3 \leq x_1 < 0.9; \\ 0, & \text{si } -0.3 \leq x_1 < 0.3; \\ -0.9, & \text{si } -0.9 \leq x_1 < -0.3; \\ -1.8, & \text{si } -1.5 \leq x_1 < -0.9; \\ -2.7, & \text{si } -2.1 \leq x_1 < -1.5; \\ -3.6, & \text{si } x_1 \leq -2.1. \end{cases} . \tag{C.8}$$

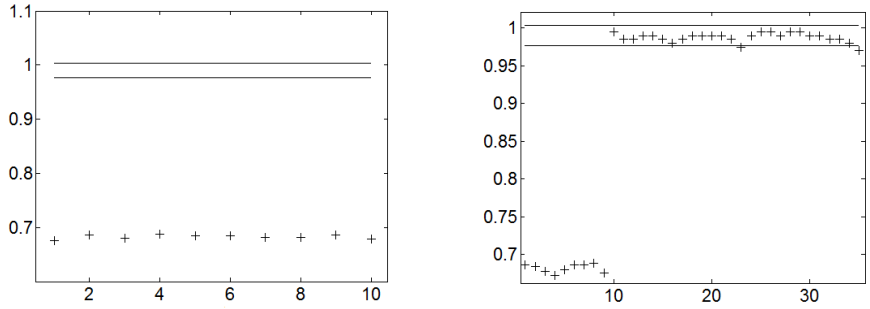
El sistema dado por la ecuación (C.3) en conjunto con la ley de conmutacion (C.8) produce 10 enroscados, la proyección en el plano (x_1, x_2) con la condición inicial $\mathbf{X}_0 = (4,0,0,1)^T$ produce el atractor de la figura C.1 c.

Los exponentes positivos de Lyapunov del sistema son (0.136181, 0.135918), indicando que el sistema es hipercaótico. Estos valores se mantienen sin importar el número de enroscados en el atractor.

C.2. Generador pseudo-aleatorio

Este generador está basado en las series de tiempo obtenidas a partir del sistema hipercaótico con multi-enroscados dado por las ecuaciones (C.2) y (C.3) para diferente número de enroscados (2, 4 o 10). La idea general es iterar el sistema n veces para obtener la secuencia $\mathbf{X}$ después de 1000 iteraciones del estado transitorio. Teniendo en cuenta la sensibilidad a las condiciones iniciales, consideramos que cada conjunto de condiciones iniciales $\mathbf{X}_{0p}$ con $p \in \mathbf{Z}^+$ resulta en p series de tiempo diferentes, donde $\mathbf{X}_{01} \neq \ldots \neq \mathbf{X}_{0p}$, en donde la llave del sistema es $\mathbf{X}_{0p}$.

El generador pseudo-aleatorio se define como se muestra a continuación:



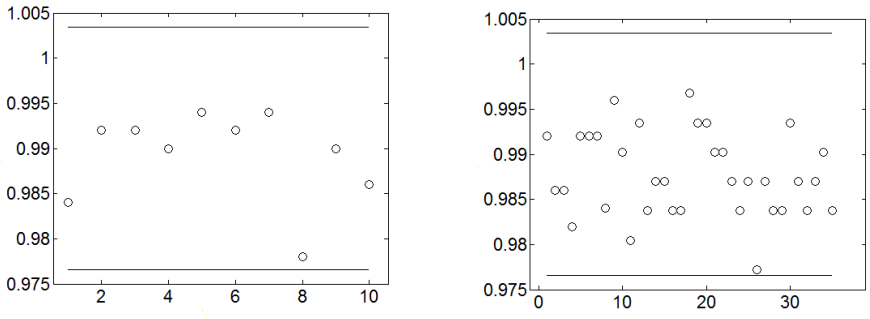
(a) Parte 1 (b) Parte 2

Figura C.2: Resultados de las pruebas estadísticas del sistema con 2 enroscados dentro del intervalo de confianza.

$$\kappa_i = \left\lfloor \sum_{j=1}^{4} x_j(i) \cdot 10^{14} \right\rfloor \ mod\ 256, \tag{C.9}$$

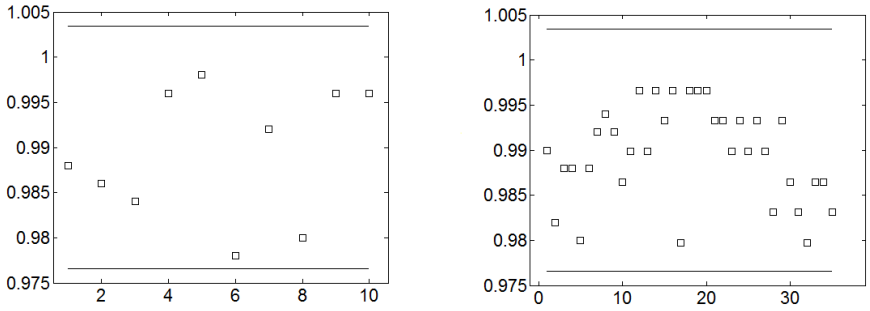
donde $\kappa_i \in \{0,1,2,\ldots,255\}$, $i = 1,\ldots,n$, y $n = l \times m$ donde l, m representan el tamaño de la imagen que se desea cifrar. Cada valor de κ_i es un número entero, el cual puede ser representado por una secuencia de 8 bits, por lo tanto tendremos una secuencia binaria de $8n = l \times m \times 8$ bits. Por otro lado la operación $\lfloor \cdot \rfloor$ representa la función piso. Para realizar estas simulaciones las variables se tomaron en formato de punto flotante con doble precisión por lo tanto tenemos cifras de hasta 10^{14} decimales.

Para evaluar este generador por medio de pruebas estadísticas se tomó una muestra de 500 secuencias con una longitud de 1,000,000 de elementos cada una. Estas pruebas estadísticas fueron aplicadas a diferentes muestras generadas con diferente número de enroscados, los resultados se presentan en las tablas C.1 y C.2 . Los resultados para una muestra de secuencias generadas con dos enroscados nos indican que no pasa 20 pruebas, en las figura C.2 se muestran los resultados dentro del intervalo de confianza, por lo tanto las secuencias generadas con dos enroscados no son confiables para su uso en criptografía. Por otro lado si el sistema presenta 4 o 10 enroscados los resultados de las pruebas estadísticas son satisfactorios en las figuras C.3 y C.4 se muestran los resultados dentro del intervalo de confianza. Por lo tanto las secuencias generadas con 4 o 10 enroscados son confiables para su uso en criptografía.



(a) Parte 1 (b) Parte 2

Figura C.3: Resultados de las pruebas estadísticas del sistema con 4 enroscados dentro del intervalo de confianza.



(a) Parte 1 (b) Parte 2

Figura C.4: Resultados de las pruebas estadísticas del sistema con 10 enroscados dentro del intervalo de confianza.

Tabla C.1: Parte 1 de los resultados del banco de pruebas estadísticas.

Prueba estadística	2 enroscados Porción de secuencias aprobadas	4 enroscados Porción de secuencias aprobadas	10 enroscados Porción de secuencias aprobadas
Frecuencia (Mono-bit)	0.6760*	0.9840	0.9880
Frecuencia dentro de un bloque (Bloque=128)	0.6860*	0.9920	0.9860
Corridas	0.6800*	0.9920	0.9840
Corrida larga	0.6880*	0.9900	0.9960
Rango de una matriz binaria	0.6840*	0.9940	0.9980
Transformada discreta de Fourier	0.6840*	0.9920	0.9760
Búsqueda de patrones sobrepuestos (Bloque=9)	0.6820*	0.9940	0.9920
Maurer	0.6820*	0.9780	0.9780
Entropía (Bloque=10)	0.6860*	0.9900	0.9960
Complejidad lineal (Bloque=500)	0.6780*	0.9860	0.9960

Tabla C.2: Parte 2 de los resultados del banco de pruebas estadísticas.

Prueba estadística	2 enroscados Secuencias aprobadas	4 enroscados Secuencias aprobadas	10 enroscados Secuencias aprobadas
Serial 1 (Bloque=16)	0.6860*	0.9920	0.9900
Serial 2 (Bloque=16)	0.6840*	0.9860	0.9820
Sumas acumulativas			
a)Hacia adelante	0.6780*	0.9860	0.9880
b)Hacia atrás	0.6720*	0.9820	0.9880
Búsqueda de patrones no sobrepuestos (Bloque=9)			
a)	0.6800*	0.9920	0.9800
b)	0.6860*	0.9920	0.9880
c)	0.6860*	0.9920	0.9920
d)	0.6880*	0.9840	0.9940
Caminatas aleatorias			
a) -4	0.9950	0.9935	0.9865
b) -3	0.9849	0.9902	0.9899
c) -2	0.9849	0.9805	0.9966
d) -1	0.9899	0.9837	0.9899
e) 1	0.9899	0.9870	0.9966
f) 2	0.9849	0.9870	0.9932
g) 3	0.9799	0.9837	0.9966
h) 4	0.9849	0.9837	0.9797
Caminatas aleatorias (Variante)			
a) -9	0.9899	0.9967	0.9966
b) -8	0.9899	0.9935	0.9966
c) -7	0.9899	0.9935	0.9966
d) -6	0.9899	0.9902	0.9932
e) -5	0.9849	0.9902	0.9932
f) -4	0.9749	0.9870	0.9899
g) -3	0.9899	0.9837	0.9932
h) -2	0.9950	0.9870	0.9899
i) -1	0.9950	0.9739	0.9932
j) 1	0.9899	0.9870	0.9899
k) 2	0.9950	0.9837	0.9831
l) 3	0.9950	0.9837	0.9932
m) 4	0.9899	0.9935	0.9865
n) 5	0.9899	0.9870	0.9831
o) 6	0.9849	0.9837	0.9797
p) 7	0.9849	0.9870	0.9865
q) 8	0.9799	0.9902	0.9865
r) 9	0.9698	0.9837	0.9831

Tabla C.3: Correlación de pixeles adyacentes en las imágenes *P* y *C* en distintas direcciones.

Imagen	Imagen original	2 enroscados	4 enroscados	10 enroscados
Lenna				
Vertical	0.9829	0.0207	0.0306	-0.0343
Horizontal	0.9687	-0.0515	0.0044	0.0017
Diagonal	0.9520	0.0284	-0.0146	0.0013
Einstein				
Vertical	0.9832	0.0450	0.0086	0.0082
Horizontal	0.9795	-0.0205	0.0168	0.0152
Diagonal	0.9677	-0.0440	-0.0258	-0.0206
Mandril				
Vertical	0.8186	0.0184	0.0179	0.0001
Horizontal	0.8641	-0.0333	0.0201	0.0055
Diagonal	0.7766	0.0253	-0.0175	0.0018

C.3. Cifrado de información

Después de haber probado que el sistema hipercaótico con cuatro o más enroscados muestra resultados satisfactorios en las pruebas de aleatoriedad del NIST, tomaremos estas secuencias para cifrar imágenes en escala de grises por medio del cifrado en flujo con retardo mostrado en la sección 4.3, el propósito de cifrar la información con este generador es mostrar que la calidad de cifrado varía de acuerdo al número de enroscados que se utilizan. Las ecuaciones que se utilizaron están dadas por la ecuación (4.11).

A continuación se muestran los resultados de las pruebas de seguridad aplicadas a las imágenes cifradas. En la tabla C.3 se muestra la correlación de pixeles adyacentes de las imágenes *P* y *C* para diferente número de enroscados. Se puede observar que en la imagen *P* existe una gran correlación de pixeles en cualquier dirección, y por otro lado en la imagen *C* los pixeles adyacentes presentan baja correlación.

La entropía la podemos calcular usando la ecuación (4.5), en la tabla C.4 se muestra la entropía para las imágenes *P* y *C*.

Tabla C.4: Entropía de las imágenes *P* y *C* cifradas con diferente número de enroscados.

Entropía	Imagen original	2 enroscados	4 enroscados	10 enroscados
Lenna	7.8059	7.9986	7.9989	7.9989
Einstein	7.7091	7.9967	7.9973	7.9973
Mandril	7.4913	7.9988	7.9989	7.9989

Tabla C.5: Calidad de cifrado para imágenes cifradas con diferente número de enroscados.

Calidad de Cifrado	2 enroscados	4 enroscados	10 enroscados
Lenna	271.1094	271.5391	271.6563
Einstein	314.4609	314.6250	314.8906
Mandril	118.6953	119.7188	119.9688

Se puede observar que en algunos casos el valor de la entropía se incrementa ligeramente al incrementar el número de enroscados mientras que en el peor de los casos la entropía se mantiene.

La calidad de cifrado la podemos calcular aplicando la ecuación (4.6), se requiere de la imagen original *P* y la imagen cifrada *C* para poder evaluar dicha prueba, en la tabla C.5 se muestran los resultados obtenidos.

Para más detalles acerca de este trabajo se puede consultar el artículo correspondiente en la referencia [179].

Productividad

Publicaciones en revistas

- García-Martínez M., Campos-Cantón I., Campos-Cantón E. Celikovský S., Difference map and its electronic circuit realization. Nonlinear Dynamics, Vol. 74, No. 3, 819-930, 2013.
- García-Martínez M., Campos-Cantón E., Pseudo-random bit generator based on lag time series. International Journal of Modern Physics C, Vol. 25, No. 4, 1350105, 2014.
- García-Martínez M., Ontañón-García L. J., Campos-Cantón E. Celikovský S., Hyperchaotic encryption on multi-scroll piecewise linear systems **enviado**.
- García-Martínez M., Campos-Cantón E., Pseudo-random bit generator based on multi-modal maps **enviado**.

Congresos internacionales

- Ontañón-García L. J., García-Martínez M., Campos-Cantón E., Celikovsky S., Grayscale image encryption using a hyperchaotic unstable dissipative system. 8th International Conference for internet technology and secured transactions, ICITST, IEEE Computer society. 6750252, 503-507, 2013.
- Real World Cryptography Workshop, 2014.
- García-Martínez M., Campos-Cantón E., Grayscale image encryption based on multimodal maps. 7th Chaotic modeling and simulation, ISBN: 978-618-81257-3-5, 2014.

Congresos nacionales

- LIV Congreso nacional de física. Poster: Implementación electrónica de mapeo caótico bimodal. 2011.
- Congreso interdisciplinario del Instituto Potosino de Investigación Científica y Tecnológica (IPICYT). Poster: Sistemas caóticos de tiempo discreto. 2011.
- LVI Congreso nacional de física. Poster: Cifrado de imágenes por medio de mapeos multi-modales. 2013.
- Congreso interdisciplinario del Instituto Potosino de Investigación Científica y Tecnológica (IPICYT). Poster: Generador pseudo-aleatorio de bits basado en mapeos. 2013.
- 10° Coloquio nacional de códigos, criptografía y áreas relacionadas. Ponencia: Cifrado de imágenes por medio de mapeos multi-modales. 2013.

Divulgación

- Conferencia “Criptografía con sistemas dinámicos discretos”, impartida en el Instituto Potosino de Investigación Científica y Tecnológica, División de Matemáticas Aplicadas, 2014.
- Artículo “La máquina enigma, mensajes cifrados para la guerra”, revista universitarios potosinos, No. 180, 2014.
- Conferencia “Encriptación y mapeos caóticos”, impartida en el simposio de matemáticas y física de la Universidad Autónoma de Aguascalientes, 2014.

Bibliografía

[1] Kahn D., The Codebreakers: The Comprehensive History of Secret Communication from Ancient Times to the Internet, Scribner Book Co, 1996.

[2] Singh S., The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography, Anchor, 2000.

[3] Vernam G. S., Cipher Printing Telegraph Systems For Secret Wire and Radio Telegraphic Communications. American Institute of Electrical Engineers, Vol. XLV, 295-301, 1926.

[4] Shannon C., Communication Theory of Secrecy Systems. Bell System Technical Journal, Vol. 28, 656-715, 1949.

[5] Biham E., Shamir A., Differential cryptanalysis of DES-like cryptosystems. Journal of Cryptology, Vol. 4, No. 1, 1-72, 1991.

[6] National Bureau of Standars: Data Encryption Standard, Federal Information Processing Standards Publication 46, US Goverment Printing Office, Whashington DC, 1977.

[7] Daemen J., Rijmen V., The Design of Rijndael, AES - The Advanced Encryption Standard. Springer, 2002.

[8] Diffie W., Hellman M.E., New directions in cryptography. IEEE Transactions on Information Theory, Vol. 22 No.6, 644–654, 1976.

[9] Rivest R. L., Shamir A., Adleman L., Method for Obtaining Digital Signatures and Public-Key Cryptosystem. Communications of the ACM, Vol 21, No. 2, 120-123, 1978.

[10] Elgamal T., A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. IEEE Transactions on Information Theory, Vol. 31, No. 4, 469-472, 1985.

[11] Koblitz N., Hyperelliptic cryptosystems. Journal of Cryptology, Vol. 1, No. 3, 139-150, 1989.

[12] Martín del Rey A., Pereira Mateus J., Rodríguez Sánchez G., A secret sharing scheme based on cellular automata. Applied Mathematics and Computation, Vol. 170, 1356-1364, 2005.

[13] Zhao Q., Yin H., Gbits/s physical-layer stream ciphers based on chaotic light. Optik, Vol. 124, No. 15, 2161-2164, 2013.

[14] Strogatz S. H., Nonlinear Dynamics and CHAOS, Perseus books, 1994.

[15] Stephen Jay Gould, The Value of Science: Essential Writings of Henri Poincare. Modern Library Science, 2001.

[16] Lorenz E., Deterministic non periodic flow. Journal of the Atmospheric Sciences, Vol. 20, 130–141, 1963.

[17] Metropolis N., Stein M., Stein P., On finite limit sets for transformations on the unit interval. Journal of Combinatorial Theory, Series A, Vol. 15, No. 1, 25–44, 1973.

[18] Li T. Y., Yorke J. A., Period three implies chaos. The American Mathematical Monthly, Vol. 82, No. 10, 985–992, 1975.

[19] May R. M., Simple mathematical models with very complicated dynamics. Nature, Vol. 261, No. 5560, 459–67, 1976.

[20] Grossmann S., Thomae S., Invariant distributions and stationary correlation functions of one-dimensional discrete processes. Zeitschrift fur Naturforschung A, Vol. 32, 1353, 1977.

[21] Mandelbrot B., Fractals: Form, Chance and Dimension, W.H.Freeman and Company, 1977

[22] Mandelbrot B., The fractal geometry of nature, W. Freeman, 1982.

[23] Feigenbaum M. J., Universal behavior in nonlinear systems. Physica D: Nonlinear Phenomena, Vol. 7, No. 1-3, 16–39, 1983.

[24] Matsumoto T., Chua L., Komuro M., The double scroll. IEEE Transactions on Circuits and Systems, Vol. 32, No. 8, 797–818, 1985.

[25] Pecora L. M., Carroll T. L., Synchronization in chaotic systems. Physical Review Letters, Vol. 64, No. 8, 821–824, 1990.

[26] Elaydi S. N., Discrete Chaos, Chapman-Hal, 2000.

[27] Lynch S., Dynamical Systems with Applications using Maple, Birkhauser-Springer, 2010.

[28] Holmgren R. A., A First Course in Discrete Dynamical Systems, Springer-Verlag, 1996.

[29] Dendrinos D. S., Sonis M., Socio-spatial stocks and antistocks; the logistic map in real space. Tha Annals of Regional Science, Vol. 27, No. 4, 197-313, 1993.

[30] Campos-Cantón E., Femat R., Pisarchik A. N., A family of multimodal dynamic maps. Communications in Nonlinear Science and Numerical Simulation, Vol. 16, No. 9, 3457-3462, 2011.

[31] Oishi S., Inoue H., Pseudo-Random Number Generators and Chaos. Transactions of the Institute of Electronics and Communication Engineers of Japan, Vol. 65, No.9, 534-541. 1982.

[32] Matthews R., On the derivation of a chaotic encryption algorithm. Cryptologia, Vol. 13, No.1, 29-42, 1989.

[33] Mitchell D. W., Nonlinear key generators. Cryptologia, Vol. 14, 250-254, 1990.

[34] González J. A., Pino R., Random number generator based on unpredictable chaotic functions. Computer Physics Communications, Vol. 120, No. 2, 109-114, 1999.

[35] Gonzalez J. A., Martin-Landrove M., Trujillo L., Absolutely unpredictable chaotic sequences. International Journal of Bifurcation and Chaos in Applied Sciences and Engineering, Vol. 10, No. 8, 1867-1874, 2000.

[36] Stojanovski T., Kocarev L., Chaos-based random number generators - Part I: Analysis. IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, Vol. 48, No. 3, 281-288, 2001.

[37] Stojanovski T., Pihl J., Kocarev L., Chaos-based random number generators - Part II: Practical realization. IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, Vol. 48, No. 3, 382-385, 2001.

[38] Andrecut M., Logistic map as a random number generator. International Journal of Modern Physics B, Vol. 12, No. 9, 921-930, 1998.

[39] Liu J., Design of a chaotic random sequence and its application. Computer Engineering, Vol. 31, No. 18, 150-152, 2005.

[40] Wang X. Y., Xie Y. X., A design of pseudo-random bit generator based on single chaotic system. International Journal of Modern Physics C, Vol. 23, No.3, 1250024, 2012.

[41] Li S., Mou X., Cai Y., Pseudo-random Bit Generator Based on Couple Chaotic Systems and Its Applications in Stream-Cipher. Progress in Cryptology — INDOCRYPT 2001, Lecture Notes in Computer Science, Vol. 2247, 316-329, 2001.

[42] Kanso A., Smaoui N., Logistic chaotic maps for binary numbers generations. Chaos, Solitons and Fractals, Vol. 40, No. 5, 2557-2568, 2009.

[43] Patidar V., Sud K. K., Pareek N. K., A pseudo random bit generator based on chaotic logistic map and its statistical testing. Informatica, Vol. 33, No. 4, 441-452, 2009.

[44] Patidar V. R., Sud K. K., A novel pseudo random bit generator based on chaotic standard map and its testing. Electronic Journal of Theoretical Physics, Vol. 6, No. 20, 327-344, 2009.

[45] Kanso A., Smaoui N., Irregularly decimated chaotic map(s) for binary digits generations. International Journal of Bifurcation and Chaos, Vol. 19, No. 4, 1169-1183, 2009.

[46] Wang X. Y., Yang L., Design of pseudo-random bit generator based on chaotic maps. International Journal of Modern Physics B, Vol. 26, No. 32, 1250208, 2012.

[47] François M., Grosges T., Barchiesi D., Erra R., Pseudo-random number generator based on mixing of three chaotic maps. Communications in Nonlinear Science and Numerical Simulation, Vol. 19, No. 4, 887-895, 2014.

[48] Menezes A. J., van Oorschot P. C., Vanstone S. A. Handbook of Applied Cryptography, CRC Press, 1997.

[49] Stinson D., Cryptography: Theory and practice, CRC Press, Boca Raton, 2005.

[50] Shannon C., A mathematical theory of communication. ACM SIGMOBILE Mobile Computing and Communications Review, Vol.5 No.1, 3–55, 2001.

[51] Kerckhoff A., La cryptographie militaire. Journal des sciences militaires, Vol. IX, 5–38, 1883.

[52] Paar C., Pelzl J., Understanding Cryptography, Springer, 2011.

[53] Elaine Barker, William Barker, William Burr, William Polk, Miles Smid, Recommendation for Key Management-Part 1: General (Revision 3). National Institute of Standards and Technology, Special Publication SP 800–57 Rev. 3a, 2012.

[54] Europan Netwok of Excellence in Cryptology, ECRYPT II Yearly Report on Algorithms and Keysizes, 2012.

[55] Stevenson F. A., Cryptanalysis of contents scrambling system, 1999.

[56] Kuznetsov Y., Elements of Applied Bifurcation Theory. Springer, 2004.

[57] Devaney R., An Introduction to Chaotic Dynamical Systems. Westview Press, 2003.

[58] Banks J., Brooks J., Cairns G., Davis G., Stacey P., On Devaney's Definition of Chaos. The American Mathematical Monthly, Vol. 99, No.4, 332-334, 1992.

[59] Vellekoop M., Berglund R., On intervals: transitivity $\rightarrow$ chaos. The American Mathematical Monthly, Vol. 101, No. 4, 353-355, 1994.

[60] Tancredi G., Sánchez A., Roig F., A Comparison Between Methods to Compute Lyapunov Exponents. The Astronomical Journal, Vol. 121, No. 2, 1171-1179, 2001.

[61] García-Martínez M., Campos-Cantón E., Pseudo-random bit generator based on lag time series. International Journal of Modern Physics C, Vol. 25, No. 4, 1350105, 2014.

[62] García-Martínez M., Campos-Cantón I., Campos-Cantón E. Celikovský S., Difference map and its electronic circuit realization. Nonlinear Dynamics, Vol. 74, No. 3, 819-930, 2013.

[63] Alvarez G., Li S., Some basic cryptographic requirements for chaos based cryptosystems. International Journal of Bifurcations and Chaos, Vol.16, No. 8, 2129-2151, 2006.

[64] Kocarev L., Chaos-based cryptohgraphy: A brief overview, IEEE Circuits and Systems Magazine, Vol.1, No. 3, 6-21, 2001.

[65] Cristian-Iulian RIncu, Alexandru Serbanescu, Chaos-based cryptography. A possible solution for information security, Bulletin of the Transilvania University of Brasov, Series III, Vol. 2, No. 51, 113-126, 2009.

[66] Mishkovski I., Kocarev L., Chaos-Based Public-key cryptography. Studies in Computational Intelligence, Vol. 354, 27-65, 2011.

[67] Cuomo, Kevin M., Oppenheim A. V., Strogatz Steven H., Synchronization of Lorenz-based chaotic circuits with applications to communications. IEEE Transactions on Circuits and Systems II, Vol. 40, No. 10, 626-633, 1993.

[68] Morgül O., Feki M., A chaotic masking scheme by using synchronized chaotic systems. Physics Letters A, Vol. 251, No. 3, 169-176, 1999.

[69] Ryabov V.B., Usik P.V., Vavriv D.M., Chaotic masking without synchronization. International Journal of Bifurcation and Chaos, Vol. 9, No. 6, 1181-1187, 1999.

[70] Xia F., Fan L., A new chaotic secure masking method of communication system (Conference paper). International Conference on Future Wireless Networks and Information Systems ICFWI, Vol. 144, No. 2, 663-667, 2012.

[71] Lau F. C. M., Yip M. M., Tse C. K., Hau S.F., A multiple-access technique for differential chaos-shift keying. IEEE Transactions on Circuits and Systems I, Vol. 49, No. 1, 96-104, 2002.

[72] Tam W. M., Lau F. C. M., Tse C. K., Generalized correlation-delay-shift-keying scheme for noncoherent chaos-based communication systems. IEEE Transactions on Circuits and Systems I, Vol. 53, No. 3, 712-721, 2006.

[73] Xu W. K., Wang L., Kolumbán G., A novel differential Chaos Shift Keying modulation scheme. International Journal of Bifurcation and Chaos, Vol.21, No. 3, 799-814, 2011.

[74] Dedieu H., Kennedy M. P., Hasler M., Chaos shift keying: modulation and demodulation of a chaotic carrier using self-synchronizing Chua's circuits. IEEE Transactions on Circuits and Systems II, Vol. 40, No. 10, 634-642, 1993.

[75] Yang T., Secure communication via chaotic parameter modulation. IEEE Transactions on Circuits and Systems I, Vol. 43, No. 9, 817-819, 1996.

[76] Chen S.-Y., Xi F., Liu Z., Multi-channel chaotic modulation for Analog-to-Information Conversion. Dianzi Yu Xinxi Xuebao/Journal of Electronics and Information Technology, Vol. 36, No. 1, 152-157, 2014.

[77] Xuan Quyen N., Van Yem V., Manh Hoang T., A chaotic pulse-time modulation method for digital communication. Abstract and Applied Analysis, 835304, 2012.

[78] Wang M.-J., Wang X.-Y., Pei B.-N., A new digital communication scheme based on chaotic modulation. Nonlinear Dynamics, Vol 67, No. 2, 1097-1104, 2012.

[79] Feldmann U., Hasler M., Schwarz W., Communication by chaotic signals: The inverse system approach. International Journal of Circuit Theory and Applications, Vol. 24, No. 5, 551-579, 1996.

[80] Alvarez-Ramirez J., Puebla H., Solis-Daun J., An inverse system approach for chaotic communications. International Journal of Bifurcation and Chaos, Vol.11, No. 5, 1411-1422, 2001.

[81] Puebla H., Alvarez-Ramirez J., Stability of inverse-system approaches in coherent chaotic communication. IEEE Transactions on Circuits and Systems I, Vol. 48, No. 12, 1413-1423, 2001.

[82] Leuciuc A., The realization of inverse system for circuits containing nullors with applications in chaos synchronization. International Journal of Circuit Theory and Applications, Vol. 26, No. 1, 1-12, 1998.

[83] Yang T., A survey of a chaotic secure communication systems. International Journal of computational cognition, Vol.2, No. 2, 81-130, 2004.

[84] Rulkov N. F., Sushchik M. M., Tsimring L. S., Abarbanel H. D. I., Generalized synchronization of chaos in directionally coupled chaotic systems. Physical Review E, Vol. 51, No. 2, 980-994, 1995.

[85] Abarbanel H. D. I., Rulkov N. F., Sushchik M.M., Generalized synchronization of chaos: The auxiliary system approach. Physical Review E, Vol. 53, No. 5, 4528-4535, 1996.

[86] Boutayeb M., Darouach M., Rafaralahy H., Generalized state-space observers for chaotic synchronization and secure communication. IEEE Transactions on Circuits and Systems I, Vol. 49, No. 3, 345-349, 2002.

[87] Yang T., Chua L.O., Impulsive stabilization for control and synchronization of chaotic systems: Theory and application to secure communication. IEEE Transactions on Cirsuits and Systems I, Vol. 44, No. 10, 976-988, 1997.

[88] Liu B., Liu X., Chen G., Wang H., Robust impulsive synchorinization of uncertain dynamical networks. IEEE Transactions on Circuits And Systems I, Vol. 52, No. 7, 1431-1441, 2005.

[89] Yang, T., Chua, L.O., Impulsive control and synchronization of nonlinear dynamical systems and application to secure communication. International Journal of Bifurcation and Chaos, Vol. 7, No. 3, 645-664, 1997.

[90] Rosa Jr. E., Ott E. Hess M.H., Transition to phase synchronization of chaos. Physical Review Letters, Vol. 80, No. 8, 1642-1645, 1998.

[91] Shuai J. W., Durand D. M., Phase synchronization in two coupled chaotic neurons. Physics Letters A, Vol. 264, No. 4, 289-297, 1999.

[92] Zheng Z., Hu G., Hu B., Phase slips and phase synchronization of coupled oscillators. Physical Review Letters, Vol. 81, No. 24, 5318-5321, 1998.

[93] Li Z., Xu D., A secure communication scheme using projective chaos synchronization. Chaos, Solitons and Fractals, Vol. 22, No. 2, 477-481, 2004.

[94] Yan J., Li C., Generalized projective synchronization of a unified chaotic system. Chaos, Solitons and Fractals, Vol. 26, No. 4, 1119-1124, 2005.

[95] Mainieri R., Rehacek J., Projective Synchronization in three-dimmensional chaotic systems. Physical Review Letters, Vol. 82, No. 15, 3042-3045, 1999.

[96] Shahverdiev E. M., Sivaprakasam S., Shore K. A., Lag Synchronization in time-delayed systems. Physics Letters A, Vol 292, No. 6, 320-324, 20022.

[97] Taherion S., Lai Y.-C., Observability of lag synchronization of coupled chaotic oscillators. Physical Review E, Vol. 59, No. 6, R6247-R6250, 1999.

[98] Li C., Liao X., Wong K.-W., Chaotic lag synchronization of coupled time-delayed systems and its applications in secure communication. Physica D, Vol. 194, No. 3-4, 197-202, 2004.

[99] Boccaletti S., Kurths J., Osipov G., Valladares D. L., Zhou, C. S., The synchronization of chaotic systems. Physics Report, Vol. 366, No. 1-2, 1-101, 2002.

[100] Addabbo T., Fort A., Rocchi S., Vignoli V., Chaos Based Generation of True Random Bits. Intelligent Computing Based on Chaos, Studies in Computational Intelligence, Vol. 184, 355-377, 2009.

[101] Addabbo T., Fort A., Rocchi S., Vignoli V., Digitized Chaos for Pseudo-random Number Generation in Cryptography. Chaos-Based Cryptography, Studies in Computational Intelligence, Vol. 354, 67-97, 2011.

[102] Kwok H. S., Tang W. K. S., A fast image encryption system based on chaotic maps with finite precision representation. Chaos, Solitons and Fractals, Vol. 32, No. 4, 1518-1529, 2007.

[103] Wang X., Wang, X., Zhao J., Zhang Z., Chaotic encryption algorithm based on alternant of stream cipher and block cipher. Nonlinear Dynamics, Vol. 63, No. 4, 587-597, 2011.

[104] Mazloom S., Eftekhari-Moghadam A.M., Color image encryption based on Coupled Nonlinear Chaotic Map. Chaos, Solitons and Fractals, Vol. 42, No. 3, 1745-1754, 2009.

[105] Liu H., Wang X., Color image encryption based on one-time keys and robust chaotic maps, Computers and Mathematics with Applications, Vol. 59, No. 10, 3320-3327, 2010.

[106] Tang G., Liao X., A method for designing dynamical S-boxes based on discretized chaotic map. Chaos, Solitons and Fractals, Vol. 23, No. 5, 1901-1909, 2005.

[107] Hussain I., Shah T., Gondal M. A., Mahmood H., An efficient approach for the construction of LFT S-boxes using chaotic logistic map. Nonlinear Dynamics, Vol. 71, No. 1-2, 133-140, 2013.

[108] Hussain I., Shah T., Mahmood H., Gondal M. A., Construction of S_8 Liu J S-boxes and their applications. Computers and Mathematics with Applications, Vol. 64, No. 8, 2450-2458, 2012.

[109] Nguyen A. P., Nguyen T. D., Determining quality of S-boxes using pseudo random sequences generated from stream ciphers. 12th International Conference on Algorithms and Architectures for Parallel Processing, ICA3PP, Vol. 7440, No. 2, 72-79, 2012.

[110] Liu Y., Tian S, Hu W, Xing C., Design and statistical analysis of a new chaotic block cipher for Wireless Sensor Networks. Communications in Nonlinear Science and Numerical Simulation, Vol. 17, No.8, 3267-327, 2012.

[111] Kocarev L., Jakimoski G., Logistic map as a block encryption algorithm. Physics Letters A, Vol. 289, No. 4-5, 199-206, 2001.

[112] Yang D., Liao X., Wang Y., Yang H., Wei P., A novel chaotic block cryptosystem based on iterating map with output-feedback. Chaos, Solitons and Fractals, Vol.41, No. 1, 505-510, 2009.

[113] Wang X.-Y., Bao X.-M., A novel block cryptosystem based on the coupled chaotic map lattice. Nonlinear Dynamics, Vol. 72, No. 4, 707-715, 2013.

[114] Zhou Y., Bao L., Chen C. L. P., Image encryption using a new parametric switching chaotic system. Signal Processing, Vol. 92, No. 11, 3039-3052, 2013.

[115] Guan Z. H., Huang F., Guan W., Chaos-based image encryption algorithm. Physics Letters A, Vol. 346, No. 1-3, 153-157, 2005.

[116] Fu C., Lin B. B., Miao Y. S., Liu X., Chen J. J., A novel chaos-based bit-level permutation scheme for digital image encryption. Optics Communications, Vol. 284, No. 23, 5415-5423, 2011.

[117] Kwok-Wo W., Image Encryption Using Chaotic Maps. Intelligent Computing Based on Chaos, Studies in Computational Intelligence, Vol. 184, 333-354, 2009.

[118] Li S., Li C., Chen G., Bourbakis N. G., Lo K.-T., A general quantitative cryptanalysis of permutation-only multimedia ciphers against plaintext attacks. Signal Processing: Image Communication, Vol. 23, No.3, 212-223, 2008.

[119] Kocarev L., Chaos-based cryptography: A brief overview. IEEE Circuits and Systems Magazine, Vol. 1, No. 3, 6-21, 2001.

[120] Kocarev L., Lian S., Chaos-Based Cryptography: Theory, Algorithms and Applications. Vol. 354 of Studies in Computational Intelligence, Springer, 2011.

[121] Alvarez G, Amigó J. M., Arroyo D., Li S., Lessons Learnt from the Cryptanalysis of Chaos-Bases Ciphers. Studies in Computational Intelligence, Vol. 354, 257-295, 2011.

[122] Amigó J. M., Chaos-Based Cryptography. Studies in Computational Intelligence, Vol. 184, 291-313, 2009.

[123] Orue A., Alvarez G., Pastor G., Romera M., Montoya F., Li S., A new parameter determination method for some double-scroll chaotic systems and its applications to chaotic cryptanalysis. Communications in Nonlinear Science and Numerical Simulation, Vol. 15, No. 11, 3471-3483.

[124] Orue A., Fernandez V., Alvarez G., Pastor G., Romera M., Montoya F., Sanchez-Avila C., Li S., Breaking a sc-cnn-based chaotic masking secure communication system. International Journal of Bifurcation and Chaos, Vol. 19, No. 4, 1329-1338, 2009.

[125] Orue A., Fernandez V., Alvarez G., Pastor G., Romera M., Li S, Montoya F., Determination of the parameters for a Lorenz system and application to break the security of two-channel chaotic cryptosystems. Physics Letters A, Vol. 372, No. 34, 5588-5592, 2008.

[126] Arroyo D., Alvarez G., Amigó J. M., Li S., Cryptanalysis of a family of self-synchronizing chaotic stream ciphers. Communications in Nonlinear Science and Numerical Simulation, Vol. 16, No. 2, 805-813, 2011.

[127] Zhang Y., Li C., Li Q., Zhang D., Shu S., Breaking a chaotic image encryption algorithm based on perceptron model. Nonlinear Dynamics, Vol. 59, No. 3, 1091-1096, 2012.

[128] Ozkaynak F., Yavuz S., Security problems for a pseudorandom sequence generator based on the Chen chaotic system. Computer Physics Communications Vol. 184, No. 9, 2178-2181, 2013.

[129] Liu N., Zheng M., Guo D., Study on the pseudorandomness and complexity of chaotic binary sequences. International Conference on Convergence Information Technology, ICCIT, Articulo número 4420585, 2007.

[130] Zheng Y. B., Song Y., Du B. X., Pan J., Ding Q., A novel detection of periodic phenomena of binary chaotic sequences. Wuli Xuebao/Acta Physica Sinica, Vol. 61, No. 23, 230501, 2012.

[131] Barash L., Shchur L. N., Periodic orbits of the ensemble of Sinai-Arnold cat maps and pseudorandom number generation. Physical Review E, Vol. 73, No. 3, 036701, 2006.

[132] Li C., Li S., Lo K. T., Breaking a modified substitution-diffusion image cipher based on chaotic standard and logistic maps. Communications in Nonlinear Science and Numerical Simulation, Vol. 16, No. 2, 837-843, 2011.

[133] Çokal C., Solak E., Cryptanalysis of a chaos-based image encryption algorithm. Physics Letters A, Vol. 373, No. 15, 1357-1360, 2009.

[134] Solak E., Rhouma R., Belghith S., Cryptanalysis of a multi-chaotic systems based image cryptosystem. Optics Communications, Vol. 283, No. 2, 232-236, 2010.

[135] Rhouma R., Solak E., Belghith S., Cryptanalysis of a new substitution-diffusion based image cipher. Communications in Nonlinear Science and Numerical Simulation, Vol. 15, No. 7, 1887-1892, 2010.

[136] Solak E., Cryptanalysis of chaotic ciphers. Studies in Computational Intelligence, Vol. 354, 227-256, 2011.

[137] Zhang Y., Xiao D., Cryptanalysis of S-box-only chaotic image ciphers against chosen plaintext attack. Nonlinear Dynamics, Vol. 72, No. 4, 751-756, 2013.

[138] Wang Y., Liao X., Xiang T., Wong K. W., Yang D., Cryptanalysis and improvement on a block cryptosystem based on iteration a chaotic map. Physics Letters A, Vol. 363, No. 4, 277-281, 2007.

[139] Li C., Li S., Alvarez G., Chen G., Lo K. T., Cryptanalysis of a chaotic block cipher with external key and its improved version. Chaos, Solitons and Fractals, Vol. 37, No. 1, 299-307, 2008.

[140] Yang J., Xiao D. Xiang T., Cryptanalysis of a chaos block cipher for wireless sensor network. Communications in Nonlinear Science and Numerical Simulation, Vol. 16, No. 2, 844-850, 2011.

[141] Amigó J. M., Kocarev L., Szczepanski J., Theory and practice of chaotic cryptography. Physics Letters A, Vol. 366, No. 3, 211-216, 2007.

[142] Masuda N., Jakimoski G., Aihara K., Kocarev L., Chaotic block ciphers: From theory to practical algorithms. IEEE Transactions on Circuits and Systems I, Vol. 53, No. 6, 1341-1352, 2006.

[143] Lee P. H., Chen Y., Pei S. C., Chen Y. Y., Evidence of the correlation between positive Lyapunov exponents and good chaotic random number sequences. Computer Physics Communications, Vol. 160, No. 3, 187-203, 2004.

[144] Arroyo D., Alvarez G., Li S., Li C., Fernandez V., Cryptanalysis of a new chaotic cryptosystem based on ergodicity. International Journal of Modern Physics B, Vol. 23, No. 5, 651-659, 2009.

[145] Arroyo D., Li C., Li S., Alvarez G., Cryptanalysis of a computer cryptography scheme based on a filter bank. Chaos, Solitons and Fractals, Vol. 41, No. 1, 410-413, 2009.

[146] Arroyo D., Rhouma R., Alvarez G., Li S., Fernandez V., On the security of a new image encryption scheme based on chaotic map lattices. Chaos, Vol. 18, No. 3, 033112, 2008.

[147] Li S., Álvarez G., Chen G., Breaking a chaos-based secure communication scheme designed by an improved modulation method. Chaos, Solitons and Fractals, Vol. 25, No. 1, 109-120, 2005.

[148] Li S., Chen G., Álvarez G., Return-map cryptanalysis revisited. International Journal of Bifurcation and Chaos, Vol. 16, No. 5, 1557-1568, 2006.

[149] Wang X., Zhan M., Lai C. H., Gang H., Error function attack of chaos synchronization based encryption schemes. Chaos, Vol. 14, No. 1, 128-137, 2004.

[150] Zhang Y., Tao C., Jiang J. J., Theoretical and experimental studies of parameter estimation based on chaos feedback synchronization. Chaos, Vol. 16, No. 4, 043122, 2006.

[151] Arroyo D., Framework fot the analysis and desing of encryption strategies based on discrete-time chaotic dynamical systems. Ph.D thesis, ETSIA of the Polytechnic University of Madrid, Madrid, Spain, 2009.

[152] Arroyo D., Alvarez G., Li S., Li C., Nunez J., Cryptanalysis of a discrete-time synchronous chaotic encryption system. Physics Letters A, Vol. 372, No. 7, 1034-1039, 2008.

[153] Álvarez G., Montoya F. Romera M., Pastor G., Cryptanalysis of an ergodic chaotic cipher. Physics Letters A, Vol. 311, No.2-3, 172-179, 2003.

[154] Rhouma R., Solak E., Arroyo D., Li S., Alvarez G., Belghith S., Comment on ”Modified Baptista type chaotic cryptosystem via matrix secret key”[Phys. Lett. A 372 (2008) 5427]. Physics Letters A, Vol. 373, No. 37, 3398-3400, 2009.

[155] Alvarez G., Montoya F., Romera M., Pastor G., Cryptanalysis of a chaotic encryption system. Physics Letters A, Vol. 276, No. 1-4, 191-196, 2000.

[156] Arroyo D., Rhouma R., Alvarez G., Li S., Fernandez V., On the security of a new image encryption scheme based on chaotic map lattices. Chaos, Vol. 18, 3, 033112, 2008.

[157] Letellier C., Gouesbet G., Topological characterization of reconstructed attractors modding out symmetries. Journal de Physique II, Vol.6 No. 11, 1615-1638, 1996.

[158] Yang T., Yang L. B., Yang C. M., Breaking chaotic secure communication using a spectrogram. Physics Letters A, Vol. 247, No. 1-2, 105-111, 1998.

[159] Kocher P. C., Timing attacks on implementations of diffie-hellman, RSA, DSS, and other systems. CRYPTO, LNCS, Vol. 1109, 104–113, 1996.

[160] Brumley D., Boneh D., Remote timing attacks are practical. Computer Networks, Vol. 48, No. 5, 701-716, 2005.

[161] Brumley B. B., Tuveri N., Remote timing attacks are still practical. 16th European Symposium on Research in Computer Security, ESORICS, Vol. 6879, 355-371, 2011.

[162] Pareek N. K., Patidar V., Sud K. K., Discrete chaotic cryptography using external key. Physics Letters A, Vol. 309, No.1-2, 75-82, 2003.

[163] Álvarez G., Montoya F., Romera M., Pastor G., Cryptanalysis of a discrete chaotic cryptosystem using external key. Physics Letters A, Vol. 319, No. 3-4, 334-339, 2003.

[164] Álvarez G., Montoya F., Romera M. Pastor G., Keystream cryptanalysis of a chaotic cryptographic method. Computer Physics Communications, Vol. 156, No. 2, 205-207, 2004.

[165] Álvarez G., Montoya F., Romera M., Pastor G., Cryptanalysis of dynamic look-up table based chaotic cryptosystems. Physics Letters A, Vol. 326, No. 3-4, 211-218, 2004.

[166] Gao T., Chen Z., Image encryption based on a new total shuffling algorithm. Chaos, Solitons and Fractals, Vol. 38, No. 1, 213-220, 2008.

[167] Alvarez G., Li S., Hernandez L., Analysis of security problems in a medical image encryption system. Computers in Biology and Medicine, Vol. 37. No. 3, 424-427, 2007.

[168] Alvarez G., Li S., Breaking an encryption scheme based on chaotic baker map. Physics Letters A, Vol. 352, No. 1-2, 78-82, 2006.

[169] Li S., Chen G., Mou X., On the dynamical degradation of digital piecewise linear chaotic maps. International Journal of Bifurcation and Chaos, Vol. 15, No. 10, 3119-3151, 2005.

[170] Beker H., Piper F., Cipher systems: The protection of communications. New York, 1982.

[171] Gustafson H., Dawson E., Nielsen L., Caelli W., A computer package for measuring the strength of encryption algorithms. Computers and Security, Vol. 13, No. 8, 687-697, 1994.

[172] Marsaglia, G., Diehard: A Battery of Tests of Randomness, http://www.stat.fsu.edu/pub/diehard/

[173] Rukhin A. et al, A Statistical test suite for random and pseudorandom number generators for cryptographic applications. NIST special publication, 800-22, 2010.

[174] Ahmed H. E. D. H., Kalash H. M., Farag Allah O. S., Encryption quality analysis of the RC5 block cipher algorithm for digital images. Optical Engineering, Vol. 45, No. 10, 107003, 2006.

[175] Addabbo T., Alioto M., Fort A., Rocchi S., Vignoli V., The digital tent map: performance analysis and optimized desing as a low-complexity source of pseudorandom bits. IEEE Transactions on Instrumentation and Measurement, Vol. 55, No. 5, 1451-1458, 2006.

[176] Campos-Cantón E., Barajas-Ramírez J. G., Solís-Perales G., Femat R. Multiscroll attractors by switching systems, Chaos, Vol. 20, No. 1, 013116, 2010.

[177] Campos-Cantón E., Femat R., Chen G., Attractors generated from switching unstable dissipative systems. Chaos, Vol. 22, No. 3, 033121, 2012.

[178] Ontañón-García L. J., Jiménez-López E., Campos-Cantón E., Basin M., A family of hyperchaotic multi-scroll attractors in Rn. Applied Mathematics and Computation, Vol. 233, 522-533, 2014.

[179] García-Martínez M., Ontañón-García L. J., Campos-Cantón E. Celikovský S., Hyperchaotic encryption on multi-scroll piecewise linear systems **enviado**.