Benemérita Universidad Autónoma de Puebla

Facultad de Ciencias de la Electrónica

Maestría en Ingeniería Electrónica,
Opción Instrumentación Electrónica

Generación de trayectorias caóticas para la planeación de rutas

Tesis
presentada como requisito parcial para obtener el grado de:

Maestro en Ingeniería Electrónica

PRESENTA:

Lic. César Hugo Pimentel Romero

Asesores:
Dr. Jesús Manuel Muñoz Pacheco
Dra. Olga Guadalupe Félix Beltrán
Dra. Luz del Carmen Gómez Pavón

*Becario CONACyT

Puebla, Pue., Septiembre 2015.

Índice general

Resumen

En el presente trabajo de tesis se describen los sistemas dinámicos no lineales con comportamiento caótico, así como sus antecedentes históricos. La idea central consiste en la búsqueda de estrategias para la planeación de rutas de exploración basadas en caos. Para ello se analizan y simulan los sistemas caóticos de Chua, Lorenz y el sistema basado en una función saturada (PWL), esta última con dos enrollamientos y con múltiples enrollamientos en 1D y 2D. Se propone una técnica para generar rutas de exploración mediante generadores de números aleatorios (RNGs). Estas señales caóticas son usadas como fuente de entropía para realizar simulaciones numéricas en MATLAB. Para diseñar los RNGs se hace un análisis de los puntos de equilibrio de cada sistema, así como del escalamiento en los niveles de excursión. Tambien se implementan algoritmos en MATLAB para analizar la dinámica caótica, desde el punto de vista de los atractores caóticos, mediante planos de Poincaré. Estos son de gran importancia en el diseño de los RNGs en la generación de rutas de exploración, ya que la aleatoriedad y la cobertura del área de exploración están fuertemente ligadas a la ubicación de estos planos.

Se diseña un RNG a partir del sistema caótico basado en el circuito de Chua y se analiza para comprobar la importancia de implementar la técnica de post-procesamiento Von Neumann en los generadores de bits aleatorios. Además se diseñan los RNGs basados en los sistema caótico de Lorenz y el sistema basado en una función saturada; como parte del análisis se diseña un RNG dual con el sistema caótico basado en una función saturada en 2D (2×2), para obtener bits a partir de dos señales del mismo sistema de manera simultánea.

Se proponen técnicas para generar rutas de exploración mediante RNGs utilizando dos sistemas caóticos a diferente frecuencia, considerando un factor de escalamiento óptimo. También se diseña un RNG híbrido combinando tres señales caóticas diferentes. Como parte del análisis se obtiene el porcentade de cobertura para cada uno de los RNGs propuestos.

Por otra parte, se determina la correlación existente entre el máximo exponente de Lyapunov y el porcentaje de cobertura en la generación de rutas de exploración para dos casos: el RNG basado en una función saturada de dos enrollamientos y para el RNG basado en el sistema de Chua normalizado. Para ello, se obtiene el máximo exponente de Lyapunov mediante series de tiempo.

Finalmente se realizan las pruebas estadísticas de aleatoriedad del paquete NIST para determinar la eficiencia de cada uno de los RNGs diseñados.

Introducción

Durante más de una década se han investigado los sistemas dinámicos no lineales con comportamiento caótico para utilizarse en una variedad de aplicaciones en diversos campos, como lo son las matemáticas, la física, la ingeniería, la economía, la sociología, etcétera [1–9].

Los sistemas caóticos poseen características especiales muy atractivas, las cuales han sido explotadas para solucionar diversos problemas en ciencias e ingeniería. Los sistemas caóticos son extremadamente sensibles a variaciones en sus condiciones iniciales, basta una pequeña diferencia entre estas para que el comportamiento futuro sea completamente diferente. Además, resulta difícil distinguirlos de un sistema aleatorio, lo que los convierte en sistemas altamente impredecibles [1, 10–14].

Por otro lado, el diseño de robots autónomos es vital para la exploración de espacios reducidos y peligrosos para la intervención humana [15, 16]. Existen muchas aplicaciones en las cuales se necesita cubrir grandes áreas, que van desde la exploración de túneles en excavaciones arqueológicas, exploración de planetas, detección de minas en misiones militares, o robots autónomos más simples como aspiradoras, cortadoras de césped o incluso juguetes [1, 13, 14, 17].

Es así como surge la idea del diseño de robots autónomos que describan trayectorias óptimas para realizar tareas de exploración, buscando maximizar la eficiencia de búsqueda mediante las características de la señal caótica. En robótica, el primer robot móvil que pudo seguir una trayectoria caótica fue propuesto por Nakamura y Seikikuchi, donde fue usada la ecuación de Arnold para generar los movimientos deseados [1]. Investigaciones posteriores en la generación de trayectorias de exploración basadas en caos se muestran en [1, 12–14, 16, 18, 19]. Sin embargo, el estudio de técnicas para la planeación de trayectorias basados en sistemas caóticos es un problema abierto.

Con base en lo anterior, el presente trabajo de tesis consiste en la búsqueda de estrategias para la planeación de rutas de exploración de un robot móvil empleando señales caóticas. Para ello se propone el análisis de los sistemas caóticos de Chua, Lorenz y el sistema basado en una función saturada, esta última con dos enrollamientos y con múltiples enrollamientos en 1D y 2D. Mediante simulaciones numéricas en MATLAB se hace un estudio de la dinámica caótica en la generación de rutas de exploración. La idea central es cubrir de manera eficiente un área determinada y que las rutas planeadas sean altamente impredecibles. Además, se determina el efecto y la dependencia del máximo exponente de Lyapunov en la planeación las rutas de exploración. Finalmente la validación de los RNGs se lleva a cabo mediante pruebas estadísticas de aleatoriedad usando paquete NIST SP800-22 [20].

La Figura 1 muestra el diagrama de bloques del enfoque del trabajo de investigación.

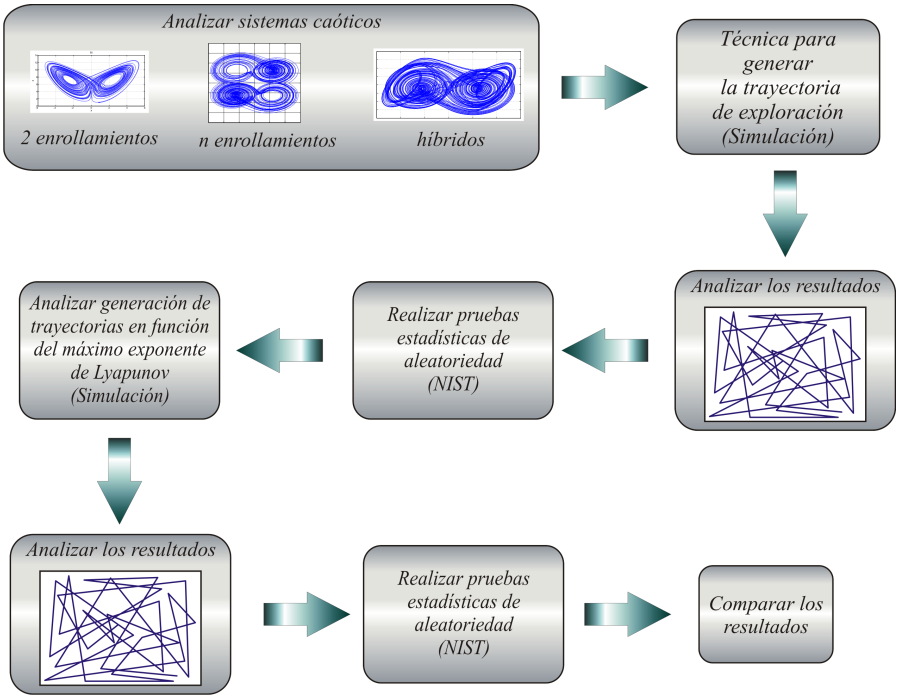


Figura 1: Diagrama a bloques del trabajo de investigación propuesto.

El objetivo general de este trabajo de tesis consiste en "Generar y analizar trayectorias de planeación de rutas empleando sistemas caóticos". Por consiguiente, los objetivos específicos se enumeran a continuación:

1. Estudiar las bases de los sistemas caóticos.
2. Proponer una técnica para la generación de trayectorias basada en una señal caótica.
3. Analizar tres sistemas dinámicos no lineales con comportamiento caótico, el sistema basado en el circuito de Chua, el sistema basado en el modelo de Lorenz y el sistema basado en una función saturada, este último con dos enrollamientos y con n enrollamientos, para generar la trayectoria de movimiento mediante simulaciones numéricas usando MATLAB.
4. Proponer y analizar una técnica de generación híbrida combinando los sistemas caóticos antes mencionados.
5. Analizar la dependencia del máximo exponente de Lyapunov en la generación de las trayectorias de exploración mediante simulaciones numéricas.
6. Analizar y comparar los resultados.

La estructura del trabajo de tesis consiste en lo siguiente: en el capítulo 1 se da una introducción general a los sistemas caóticos, mientras que la descripción y simulación de dichos sistemas se presentan en el capítulo 2. La dinámica caótica y su caracterización mediante secciones de Poincaré es tratada en el capítulo 3. En los capítulo 4 y 5 se describen las técnicas

utilizadas para la generación de rutas de exploración y los resultados obtenidos, respectivamente. El análisis de resultados es presentado en el capítulo 6 y las conclusiones son dadas en el capítulo 7. Finalmente, cinco apéndices con información adicional son incluidos.

Capítulo 1

Introducción a los sistemas caóticos

En este capítulo se presentan algunos antecedentes históricos de los sistemas caóticos, así como una breve descripción de los mismos. También se definen algunos conceptos básicos relevantes para la comprensión de la dinámica de los sistemas caóticos.

1.1. Antecedentes de los sistemas caóticos

El primer intento por describir la realidad física de una manera cuantitativa se remonta a los pitagóricos, con su esfuerzo para explicar el mundo tangible por medio de números enteros. Desde un punto de vista conceptual, el principal legado de Galileo y Newton es la idea de que la naturaleza obedece a leyes inmutables que se pueden formular en el lenguaje matemático, los eventos físicos se pueden predecir con certeza (determinismo).

Irónicamente, el primer ejemplo claro de lo que hoy conocemos como caos fue encontrado en la mecánica celeste, la ciencia de los fenómenos regulares y predecibles por excelencia. Teniendo en cuenta la ley de la gravedad, las posiciones y las velocidades iniciales de tres cuerpos que interactúan gravitacionalmente, por ejemplo, Luna-Tierra-Sol, las ecuaciones de la mecánica determinan las posiciones y velocidades posteriores. Sin embargo a pesar de la naturaleza determinista del sistema, H. Poincaré encontró que la evolución de estos cuerpos celestes puede ser de naturaleza caótica, lo que significa que pequeñas perturbaciones en el estado inicial, como un ligero cambio en la posición inicial de un solo cuerpo, podrían conducir a las diferencias dramáticas en los estados posteriores del sistema [21].

La profunda implicación de estos resultados es que el determinismo y la previsibilidad son problemas distintos. Ahora bien, los descubrimientos de Poincaré no recibieron la debida atención durante un largo tiempo. Probablemente, hay dos razones principales para tal demora. En primer lugar, a principios del siglo XX, los científicos y los filósofos perdieron interés en la mecánica clásica porque fueron atraídos principalmente por dos nuevas teorías revolucionarias: la relatividad y la mecánica cuántica. En segundo lugar, un papel importante en el reconocimiento de la importancia y ubicuidad del *caos* ha sido interpretado por el desarrollo de la computadora, que llegó mucho después de la contribución de Poincaré. De hecho, sólo gracias a la llegada de la visualización por computadora fue posible calcular (numéricamente) y ver la complejidad de comportamientos caóticos que emergen de sistemas deterministas no lineales.

Una opinión generalizada sostiene que la línea de la investigación científica abierta por Poin-

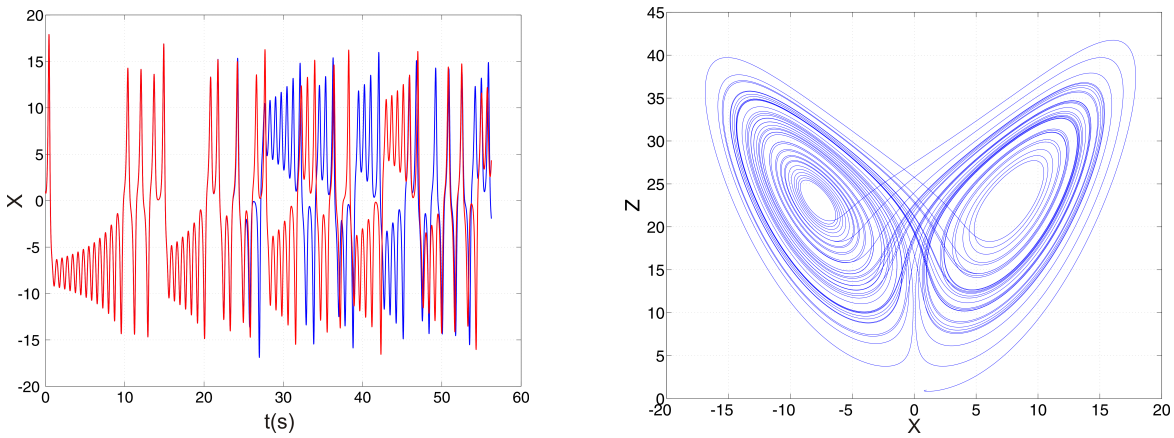
caré fue descuidada hasta 1963, cuando el meteorólogo estadounidense E. Lorenz redescubrió el caos determinista mientras estudiaba la evolución de un modelo simplificado de la atmósfera (ver ecuación (1.1)) [21].

$$\begin{aligned} \dot{x} &= \sigma(y - x), \\ \dot{y} &= \gamma x - y - xz, \\ \dot{z} &= xy - bz. \end{aligned} \tag{1.1}$$

Lorenz utilizó integración numérica para observar la evolución temporal de las señales del sistema. Estudió el sistema para un caso particular en el valor de sus parámetros e integró con una condición inicial. Al graficar, observó que después del primer transitorio, la solución presentaba una oscilación irregular que persistía en $t \to \infty$, pero nunca se repetía exactamente, un movimiento **aperiódico**. También observó que el sistema era extremadamente sensible a la variación en las condiciones iniciales.

La Figura 1.1(a) muestra la evolución temporal de la señal X del sistema de Lorenz con dos condiciones iniciales diferentes (una diferencia de 1×10^{-6} una de la otra).

Lorenz descubrió también que una maravillosa estructura emerge si la solución se visualiza como una trayectoria en el espacio de fase, esto es, cuando se grafica Z contra X, se obtiene un elegante patrón en forma de mariposa como se puede observar en la Figura 1.1(b) [11].



(a) Variable $x(t)$ con dos condiciones iniciales diferentes.

(b) Atractor del sistema de Lorenz.

Figura 1.1: Sistema de Lorenz.

1.2. Los sistemas caóticos

El caos hace referencia a un tipo de comportamiento dinámico que posee algunas características muy especiales, como lo son: *i)* extrema sensibilidad a pequeñas variaciones en sus condiciones iniciales, *ii)* trayectorias delimitadas en el espacio de fase con un exponente de Lyapunov positivo, *iii)* una entropía de Komogorov-Sinai finita, *iv)* un espectro de potencia continuo, y *v)* una dimensión topológica fraccional, entre otras [22]. En otras palabras, el **caos** *es un comportamiento aperiódico a largo término de un sistema determinístico que exhibe dependencia sensitiva a las condiciones iniciales* [11].

1.2.1. Sistemas dinámicos

Un sistema dinámico se caracteriza por un conjunto de variables relacionadas entre sí que pueden depender del tiempo de tal forma que, al menos al principio, es previsible mientras que la influencia externa sea conocida [22].

1.2.2. Representación en el espacio de estados

Con el fin de entender el comportamiento de un sistema se requiere de un modelo matemático, el cual puede ser formulado de distintas maneras, con la característica principal que nos permita conocer el comportamiento futuro del sistema, dadas las condiciones iniciales y las fuerzas externas conocidas que interactúan con el mismo [11]. La estructura matemática más natural para este propósito es la representación en variables de estado, que consiste en un conjunto de ecuaciones diferenciales, las cuales describen la evolución de las variables, cuyos valores, en un instante dado, determinan el estado actual del sistema. Dichas variables son conocidas como variables de estado y sus valores en un instante de tiempo contienen la información suficiente para que la evolución futura del sistema pueda ser determinado, dado que las interacciones externas (o variables de entrada) que actúan sobre este se conocen. Por lo tanto, la ecuación diferencial es de primer orden con respecto a la derivada del tiempo, así que los valores iniciales deben de ser suficientes para determinar la solución.

Por conveniencia de notación es común que las variables de estado se agrupen en un vector $\mathbf{x}$ (el vector de estados), las variables de entrada dentro de un vector $\mathbf{u}$ y $\mathbf{y}$ el vector de salida [22]. Entonces, las ecuaciones se expresan de la siguiente manera:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}, t), \\ \mathbf{y} &= \mathbf{h}(\mathbf{x}, \mathbf{u}, t), \end{aligned} \tag{1.2}$$

donde el punto denota diferenciación con respecto al tiempo (t), las funciones $\mathbf{f}$ y $\mathbf{h}$ son en general no lineales. Las funciones no lineales pueden surgir en un modelo dinámico debido a que son intrínsecos a la naturaleza del sistema o porque fueron introducidas deliberadamente por el diseñador para un propósito específico. La variedad de posibles no-linealidades es infinito, sin embargo, pueden ser clasificadas en algunas categorías generales. En primer lugar, hay funciones analíticas simples, tales como sinusoides y exponenciales de una sola variable, o productos de diferentes variables. Una característica importante de estas funciones es que son lo suficientemente suaves para poseer expansiones de Taylor convergentes en todos los puntos y, por tanto, puede ser linealizado [11]. Un tipo de funciones no lineales utilizadas frecuentemente en el modelado de sistemas son las funciones lineales a tramos (PWL, por sus siglas en inglés)[1] [22–25], las cuales consisten en un conjunto de relaciones lineales válidas en diferentes regiones. Tienen la ventaja de cambiar de una dinámica no lineal a ecuaciones lineales (y por lo tanto, tienen solución) en cualquier región particular, las soluciones para diferentes regiones pueden unirse entre sí en los límites. Aunque el objetivo es el estudio de sistemas dinámicos no lineales, es conveniente en este punto revisar el caso de los sistemas lineales, esto debido a que las aproximaciones lineales son ampliamente aplicables para resolver los sistemas no lineales [22].

[1] PWL, de la lengua inglesa *piecewise linear*.

Se define a un sistema lineal invariante en el tiempo en el espacio vectorial de estados como:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \\ \mathbf{y} &= \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}, \end{aligned} \tag{1.3}$$

donde $\mathbf{x}$ es el vector de estado, $\mathbf{u}$ es vector de entradas, $\mathbf{y}$ es el vector de salidas, $\mathbf{A}$ es la matriz del sistema, $\mathbf{B}$ es la matriz de entrada, $\mathbf{C}$ es la matriz de salida, $\mathbf{D}$ es la matriz de perturbación; $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, y $\mathbf{D}$ posiblemente dependan del tiempo. La gran ventaja de la linealidad es que, incluso en el caso dependiente del tiempo, puede ser construida una solución formal aplicable para todas las condiciones iniciales y todas las funciones de entrada [26]. Un punto importante que se debe tener en cuenta para un sistema dinámico no lineal es que las propiedades de estabilidad son más complicadas que en el caso lineal, esto debido a que cuando las no linealidades están presentes, pueden aparecer algunas características tales como ciclos límite o el fenómeno conocido como **caos** [11]. En cualquier caso, el tipo de comportamiento de un sistema dinámico no lineal, ya sea estable, inestable, oscilatorio o caótico, puede depender críticamente de la entrada aplicada. Por lo tanto, difiere del caso lineal donde todas las propiedades dinámicas se pueden describir, por ejemplo, por una función de transferencia, independientemente de la entrada.

1.2.3. Sistemas autónomos

Aunque las ecuaciones de un modelo dinámico en general dependerán del tiempo, ya sea de forma explícita o por medio de una función de entrada, una gran parte de la teoría de los sistemas lineales se refiere a casos donde no existe dependencia explícita del tiempo [27]. Estos sistemas se conocen como sistemas autónomos y surgen con toda naturalidad en la práctica. La ecuación diferencial en el espacio de estados se expresa como

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x} + \hat{\mathbf{u}}), \tag{1.4}$$

donde $\hat{\mathbf{u}}$ es un vector constante. Así, los puntos de equilibrio en el espacio de estados son determinados por $\mathbf{f}(\mathbf{x}, \hat{\mathbf{u}}) = 0$. Asumiendo que $\mathbf{f}(\mathbf{x}, \hat{\mathbf{u}})$ satisface la condición de Lipschitz [28], la ecuación diferencial para $\mathbf{x}(t)$ debe tener una única solución para cualquier estado inicial dado $\mathbf{x}(0)$. La trayectoria trazada en el espacio por $\mathbf{x}(t)$ se denomina trayectoria del sistema; debido a la propiedad de unicidad habrá una y sólo una trayectoria que pasa a través de cualquier punto dado. Si se elimina la dependencia en $\hat{\mathbf{u}}$, las ecuaciones diferenciales del espacio de estados para un sistema autónomo se puede escribir como [22]:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}). \tag{1.5}$$

El conjunto de todas las trayectorias de la ecuación (1.5) proporcionan una forma completa de la geometría no lineal del sistema, bajo las condiciones especificadas. Como resultado, es posible dar una clasificación completa del comportamiento en el plano de fase, aunque no en las dimensiones superiores del espacio de estados. En general todas las ecuaciones que describen el comportamiento dinámico no lineal no pueden ser resueltas de forma analítica, así que, para la construcción de las trayectorias es necesario el uso de métodos numéricos [22, 29].

1.2.4. Puntos de equilibrio

Los puntos de equilibrio de un sistema autónomo dado por la ecuación (1.5) también se conocen como puntos singulares cuando $\mathbf{f}(\dot{\mathbf{x}}) = 0$, esto porque parecen quebrantar la regla general de que sólo una trayectoria puede pasar a través de cualquier punto dado[2].

Suponiendo que $\mathbf{f}(\mathbf{x})$ es lo suficientemente suave para ser linealizada alrededor del punto singular $\hat{\mathbf{x}}$, la aproximación deberá ser suficiente para determinar el comportamiento de las trayectorias en la vecindad del punto de equilibrio. Si $\mathbf{A}$ en la ecuación (1.3) es no singular, la naturaleza del punto de equilibrio es esencialmente determinada por sus valores propios (Tabla 1.1), los cuales pueden ser clasificados en nodo estable, foco estable, nodo inestable, foco inestable, centro y punto de silla (Figura 1.2) [11, 21].

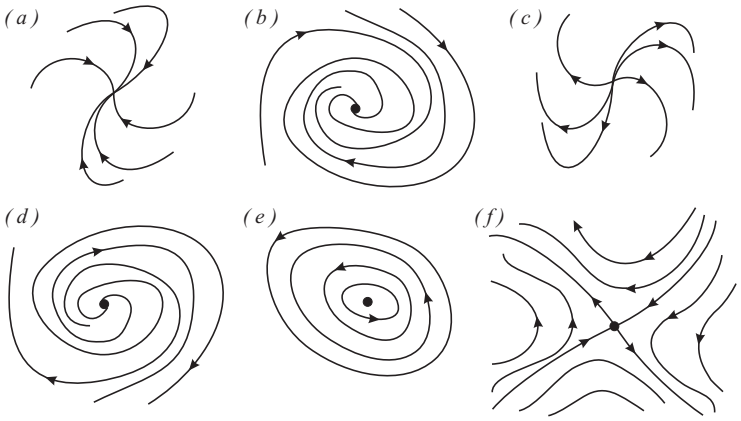


Figura 1.2: Clasificación de los puntos de equilibrio [21]

.

Clase	Eigenvalores	Tipo de punto de equilibrio
(a)	$\lambda_1 < \lambda_2 < 0$	Nodo estable
(b)	$\lambda_{1,2} = \mu \pm i\omega$ & $\mu < 0$	Foco estable
(c)	$\lambda_1 > \lambda_2 > 0$	Nodo inestable
(d)	$\lambda_{1,2} = \mu \pm i\omega$ & $\mu > 0$	Foco inestable
(e)	$\lambda_1 < 0 < \lambda_2$	Centro
(f)	$\lambda_{1,2} = \pm i\omega$	Punto de silla

Cuadro 1.1: Puntos de equilibrio y su clasificación en función a sus eigenvalores [21].

[2]En realidad es sólo aparente, ya que las trayectorias que se encuentran en un punto singular no pasan a través de él, sólo se acercan o apartan asintóticamente [22].

1.2.5. Atractores extraños y caos

El término atractor es difícil de definir de manera rigurosa, pero en términos generales, un **atractor** es un conjunto límite al que todas las trayectorias vecinas convergen cuando $t \longrightarrow \infty$, ya que atrae asintóticamente trayectorias cercanas al el mismo [21]. Un sistema autónomo de tiempo continuo requiere más de dos dimensiones para exhibir caos, en las cuales el comportamiento de las trayectorias es más compleja, ya que puede o no haber atracción asintótica para las trayectorias vecinas. A estos se les conoce como atractores extraños. Mas aún, las trayectorias que contienen podrían ser localmente divergentes entre sí dentro del conjunto de la atracción. Tales estructuras están asociadas con el comportamiento cuasi-aleatorio de soluciones llamado caos [22].

1.2.6. Representación en fase de un sistema dinámico

Un método para el análisis de las oscilaciones de los sistemas dinámicos consiste en su representación gráfica en el espacio de fase, el cual se introdujo a la teoría de las oscilaciones por L. I. Mandelstam y A.A. Andronov [30]. Desde entonces, este método se ha convertido en la herramienta habitual para el estudio de diversos fenómenos oscilatorios. Cuando las oscilaciones de forma compleja se descubrieron, por ejemplo el caos dinámico, este método aumentó en importancia. El análisis de los diagramas de fase de los procesos oscilatorios complejos permite juzgar la estructura topológica en un límite caótico establecido, así como hacer conjeturas y suposiciones que pueden ser valiosas cuando se realizan nuevas investigaciones [31]. La forma general de un campo vectorial en el plano de fase es:

$$\begin{aligned} \dot{x}_1 &= f_1(x_1, x_2), \\ \dot{x}_2 &= f_2(x_1, x_2), \end{aligned} \tag{1.6}$$

donde f_1 y f_2 son funciones dadas. Este sistema puedes ser escrito en forma más compacta en notación vectorial como:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}), \tag{1.7}$$

donde $\mathbf{x} = (x_1, x_2)$ y $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}))$. Aquí $\mathbf{x}$ representa un punto en el espacio de fase, y $\dot{\mathbf{x}}$ es el vector velocidad en ese punto. Fluyendo a lo largo del campo vectorial, un punto de fase traza una solución $\mathbf{x}(t)$, correspondiente a una trayectoria de enrollamiento a través del plano de fase. Además, todo el plano de fase está lleno de trayectorias, puesto que cada punto puede desempeñar el papel de una condición inicial.

Para los sistemas no lineales, normalmente no hay forma de encontrar las trayectorias analíticamente, incluso cuando se dispone de fórmulas explícitas. En su lugar se determina el comportamiento cualitativo de las soluciones. Entonces, el retrato fase del sistema se obtiene directamente de las propiedades de $\mathbf{f}(\mathbf{x})$, por lo que es posible generar una gran variedad de retratos de fase. En la Figura 1.3 se muestra el atractor de Lorenz (línea negra). Asimismo podemos observar de forma independiente la evolución temporal de x (línea roja) y de z (línea azul), conceptualizando de forma gráfica la definición de retrato de fase.

Algunas de las características más importantes de cualquier retrato de fase son [11]:

1. **Los puntos de equilibrio**, como A, B y C de la Figura 1.4. Recordemos que los puntos de equilibrio satisfacen $\mathbf{f}(\mathbf{x}) = 0$ y corresponden a estados estacionarios o de equilibrio del sistema.

2. **Las órbitas cerradas**, como D, corresponden a soluciones periódicas.

3. **La disposición de las trayectorias cerca de los puntos de equilibrio y las trayectorias cerradas**. Por ejemplo, el patrón de flujo cerca de A y C es similar, pero diferente de la que se encuentra cerca de B.

4. **La estabilidad o inestabilidad de los puntos de equilibrio y órbitas cerradas**. Aquí, los puntos de equilibrio A, B, y C son inestables, ya que las trayectorias cercanas tienden a alejarse de ellos, mientras que la órbita cerrada D es estable.

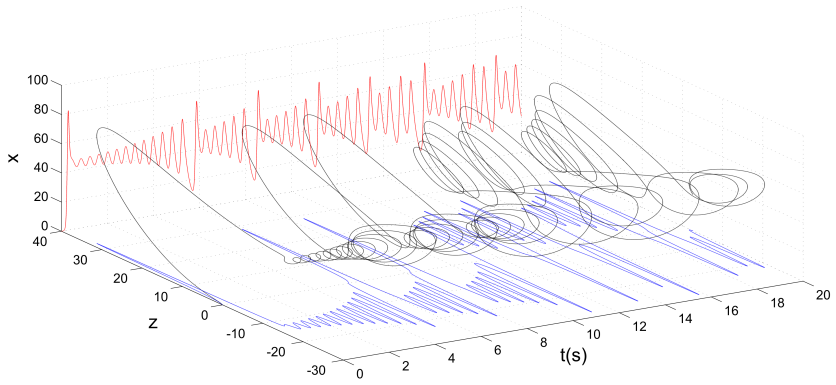


Figura 1.3: Retrato de fase del sistema de Lorenz.

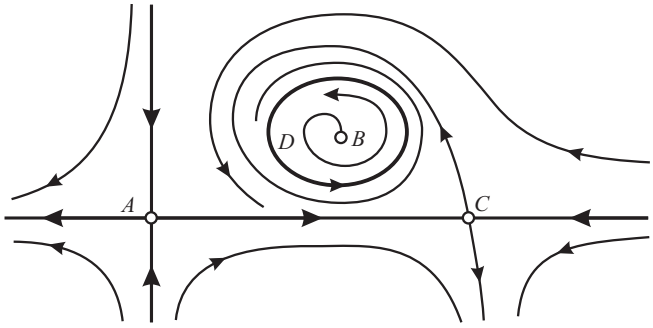


Figura 1.4: Elementos del retrato de fase.

1.2.7. Mapa de Poincaré

La visualización de las trayectorias para $d > 3$ no es posible, pero se puede recurrir a la técnica llamada mapa de Poincaré, cuya construcción se puede hacer de la siguiente manera:

- Para simplificar la representación, se considera un sistema autónomo de tres dimensiones $\mathbf{f(x)}$, y se centran en una de sus trayectorias.
- Se define un plano (en general una superficie $(d-1)$) y se consideran todos los puntos $P-n$ en que la trayectoria cruza el plano desde el mismo lado, como se ilustra en la Figura 1.5.

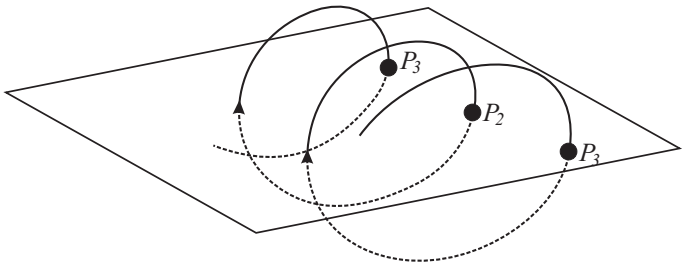


Figura 1.5: Mapa de Poincaré de una trayectoria genérica para los primeros tres puntos de intersección P_1, P_2 y P_3.

El mapa de Poincaré del flujo $\mathbf{f}$ se define entonces como el mapa $\mathbf{G}$ asociando dos puntos de cruce sucesivos, es decir:

$$\mathbf{P}_{n+1} = \mathbf{G}(\mathbf{P}_n), \tag{1.8}$$

que se puede obtener simplemente mediante la integración de la ecuación diferencial ordinaria original desde el momento de la $n-$*ésima* intersección a la $(n+1)$, por lo que siempre está bien definida. En realidad también su inversa $P_{n-1} = \mathbf{G}^{-}1(\mathbf{P}_n)$ está bien definida simplemente integrando hacia atrás, por lo tanto, el mapa de la ecuación (1.8) es invertible.

Los mapas de Poincaré permiten un espacio de fase d-dimensional para ser reducido a una representación $(d-1)$, lo que permite identificar la periodicidad (si la hay) de una trayectoria. Tales mapas también son valiosos para un análisis más refinado que solamente la visualización, porque preserva las propiedades de estabilidad de los puntos y las curvas. Es necesario señalar que la construcción de un mapa de Poincaré apropiado para un sistema genérico no es una tarea fácil, ya que la elección de un buen plano o superficie $(d-1)$ de intersección requiere experiencia [21].

Capítulo 2

Análisis de los sistemas caóticos

En este capítulo se presenta una descripción de los sistemas caóticos que se van a utilizar en este trabajo de tesis, así como la simulación numérica de cada uno de ellos: El sistema basado en el circuito de Chua, Sistema de Lorenz y sistema basado en una función saturada. También se realiza el análisis de los puntos de equilibro de estos sistemas y el escalamiento en amplitud de las señales.

2.1. Sistema basado en el circuito de Chua

El circuito de Chua (Figura 2.1) consta de cinco elementos: una resistencia lineal (R), un inductor (L), dos capacitores ($C_1 y C_2$) y una resistencia no lineal conocida como el diodo de Chua (N_R).

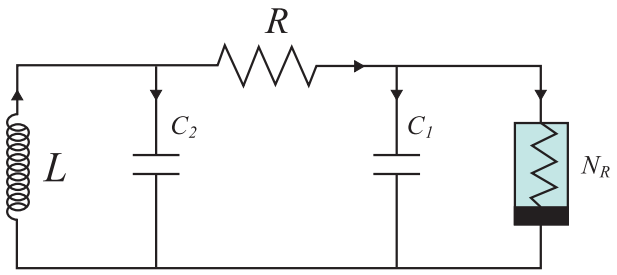


Figura 2.1: Circuito de Chua.

2.1.1. Circuito de Chua

El circuito de Chua puede ser modelado por medio de variables de estado de la siguiente forma:

- El número de variables de estado se encuentra determinado en función del número de elementos dinámicos (capacitores e inductores).

- Las variables de estado dependen únicamente de las mismas variables de estado y de las fuentes de excitación.
- Para los capacitores $C_i \longrightarrow V_{Ci}$ es la variable de estado.
- Para el inductor $L \longrightarrow i_L$ es la variable de estado.

Para determinar V_{Ci}, se busca i_{Ci} (usando LCK[1]):

$$\begin{aligned} i_R &= i_{C1} + i_{NR}, \\ \frac{V_{C2} - V_{C1}}{R} &= C_1 \frac{dV_{C1}}{dt} + i_{NR}, \\ \frac{dV_{C1}}{dt} &= -\frac{V_{C1}}{RC_1} + \frac{V_{C2}}{RC_1} - \frac{i_{NR}}{C_1}, \end{aligned}$$

y

$$\begin{aligned} i_L &= i_{C2} + i_R, \\ i_L &= C_2 \frac{dV_{C2}}{dt} + \frac{V_{C2} - V_{C1}}{R}, \\ \frac{dV_{C2}}{dt} &= \frac{V_{C1}}{RC_2} - \frac{V_{C2}}{RC_2} - \frac{i_L}{C_2}. \end{aligned}$$

Para determinar i_L, se busca V_L (usando LVK[2]):

$$\begin{aligned} V_L &= -V_{C2}, \\ L\frac{dI_L}{dt} &= -V_{C2}, \\ \frac{dI_L}{dt} &= -\frac{V_{C2}}{L}, \end{aligned}$$

donde $i_L \longrightarrow I_L$. Como resultado, el sistema queda definido de la siguiente manera:

$$\begin{aligned} \frac{dV_{C1}}{dt} &= -\frac{V_{C1}}{RC_1} + \frac{V_{C2}}{RC_1} - \frac{I_{NR}}{C_1}, \\ \frac{dV_{C2}}{dt} &= \frac{V_{C1}}{RC_2} - \frac{V_{C2}}{RC_2} - \frac{i_L}{C_2}, \\ \frac{dI_L}{dt} &= -\frac{V_{C2}}{L}. \end{aligned} \tag{2.1}$$

i_{NR} es una función no lineal característica del diodo de Chua, que puede ser linealizada con una función PWL, tal como se ilustra en la Figura 2.2. La función i_{NR} tiene la forma general dada por la ecuación (2.2), donde $m(g1, g2, g3)$ e I_x evolucionan de acuerdo a la función (2.3). Además, las ecuaciones (2.1)-(2.3) conducen a los tres sistemas de ecuaciones diferenciales de primer orden de la forma $\dot{\mathbf{x}} = \mathbf{A}x + \mathbf{B}u$, dados en las ecuaciones (2.4)-(2.6) [22].

$$i_{NR} = mV_{C1} + Ix, \tag{2.2}$$

[1]LCK: Ley de corrientes de Kirchhoff
[2]LVK: Ley de voltajes de Kirchhoff

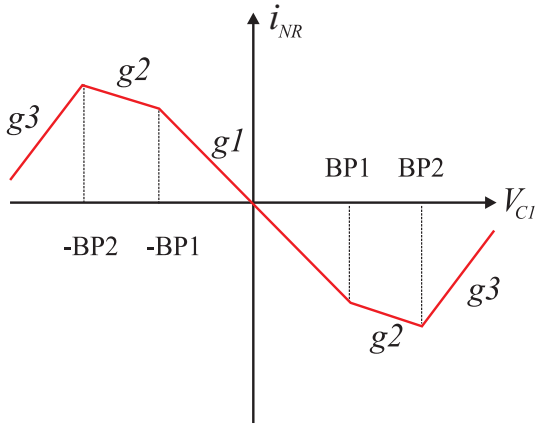


Figura 2.2: Gráfica I-V de la función PWL del diodo de Chua.

$$i_{NR} = \begin{cases} -g2V_{C1} + (g1 - g2)BP_1, & V_{C1} < -BP_1, \\ -g1V_{C1}, & -BP_1 \leq V_{C1} \leq BP_1, \\ -g2V_{C1} + (g2 - g1)BP_1, & V_{C1} > BP_1. \end{cases} \tag{2.3}$$

donde $g1$, $g2$ y $g3$ son las pendientes que se comportan como resistencias negativas; $\pm BP1$ y $\pm BP2$ representan los puntos de quiebre (ver Figura 2.2).

$$\begin{bmatrix} \dot{V}_{C1} \\ \dot{V}_{C2} \\ \dot{I}_L \end{bmatrix} = \begin{bmatrix} -\frac{1}{RC_1} + \frac{g2}{C_1} & \frac{1}{RC_1} & 0 \\ \frac{1}{RC_2} & -\frac{1}{RC_2} & \frac{1}{C_2} \\ 0 & \frac{1}{L} & 0 \end{bmatrix} \begin{bmatrix} V_{C1} \\ V_{C2} \\ I_L \end{bmatrix} + \begin{bmatrix} \frac{(g2-g1)BP_1}{C_1} \\ 0 \\ 0 \end{bmatrix}, V_{C1} < -BP_1, \tag{2.4}$$

$$\begin{bmatrix} \dot{V}_{C1} \\ \dot{V}_{C2} \\ \dot{I}_L \end{bmatrix} = \begin{bmatrix} -\frac{1}{RC_1} + \frac{g1}{C_1} & \frac{1}{RC_1} & 0 \\ \frac{1}{RC_2} & -\frac{1}{RC_2} & \frac{1}{C_2} \\ 0 & \frac{1}{L} & 0 \end{bmatrix} \begin{bmatrix} V_{C1} \\ V_{C2} \\ I_L \end{bmatrix}, -BP_1 \leq V_{C1} \leq BP_1, \tag{2.5}$$

$$\begin{bmatrix} \dot{V}_{C1} \\ \dot{V}_{C2} \\ \dot{I}_L \end{bmatrix} = \begin{bmatrix} -\frac{1}{RC_1} + \frac{g2}{C_1} & \frac{1}{RC_1} & 0 \\ \frac{1}{RC_2} & -\frac{1}{RC_2} & \frac{1}{C_2} \\ 0 & \frac{1}{L} & 0 \end{bmatrix} \begin{bmatrix} V_{C1} \\ V_{C2} \\ I_L \end{bmatrix} + \begin{bmatrix} \frac{(g1-g2)BP_1}{C_1} \\ 0 \\ 0 \end{bmatrix}, V_{C1} > BP_1. \tag{2.6}$$

2.1.2. Puntos de equilibrio del sistema basado en el circuito de Chua

El análisis de los sistemas no lineales con comportamiento caótico es complicado debido a la extrema sensitividad que presentan a cualquier variación en las condiciones iniciales, sin embargo, aunque el sistema presente diferentes soluciones, la trayectoria siempre va a evolucionar en función de la atracción a los puntos de equilibrio, también conocidos como puntos singulares

cuando $\mathbf{f}(\dot{\mathbf{x}}) = 0$, de tal modo que, a partir del sistema de ecuaciones diferenciales (2.1), se obtiene:

$$\begin{aligned} -\frac{V_{C1}}{RC_1} + \frac{V_{C2}}{RC_1} - \frac{i_{NR}}{C_1} &= 0, \\ \frac{V_{C1}}{RC_2} - \frac{V_{C2}}{RC_2} - \frac{i_L}{C_2} &= 0, \\ -\frac{V_{C2}}{L} &= 0. \end{aligned}$$

De la última ecuación se deduce que $V_{C2} = 0$, por lo tanto:

$$\begin{aligned} -\frac{V_{C1}}{R} &= i_{NR}, \\ \frac{V_{C1}}{R} &= i_L. \end{aligned}$$

Ahora bien, sustituyendo i_{NR} en la ecuación (2.2) se tiene:

$$\begin{aligned} -\frac{V_{C1}}{R} - (mV_{C1} + I_x) &= 0, \\ -\frac{V_{C1}}{R} - mV_{C1} - I_x &= 0, \\ -V_{C1}\left(\frac{1}{R} + m\right) &= I_x, \\ -V_{C1}\left(\frac{1+mR}{R}\right) &= I_x, \\ V_{C1} &= -\frac{I_x R}{1+mR}, \\ i_L &= \frac{-I_x}{1+mR}. \end{aligned}$$

Finalmente, sustituyendo I_x en las últimas dos ecuaciones se obtiene:

$$\begin{aligned} V_{C1} &= \frac{(g2-g1)BP_1R}{1+mR}, \qquad V_{C1} = -\frac{(g2-g1)BP_1R}{1+mR}, \\ i_L &= \frac{(g1-g2)BP_1}{1+mR}, \qquad i_L = -\frac{(g1-g2)BP_1}{1+mR}. \end{aligned}$$

Por lo tanto, los puntos de equilibrio quedan de la siguiente manera:

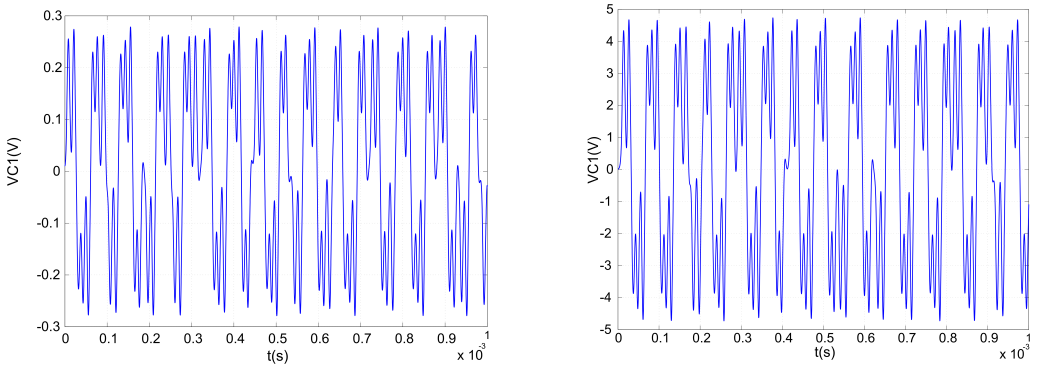
$$\begin{aligned} P1 &= (0,0,0), \\ P2 &= \left(\frac{(g2-g1)BP_1R}{1+mR}, 0, -\frac{(g1-g2)BP_1}{1+mR}\right), \\ P3 &= \left(-\frac{(g2-g1)BP_1R}{1+mR}, 0, \frac{(g1-g2)BP_1}{1+mR}\right). \end{aligned}$$

2.1.3. Escalamiento de las señales

Cabe señalar que, pensando en una futura implementación en hardware, es necesario escalar los niveles de excursión (ELs, por sus siglas en inglés) de la función PWL para que consecuentemente se escalen los ELs de las señales caóticas, de no hacerlo existen limitaciones en los rangos dinámicos de los circuitos disponibles. Por ejemplo en el caso del circuito de Chua, los valores nominales producen ELs muy pequeños en las señales, por lo tanto para obtener ELs de mayor amplitud, es necesario añadir un parámetro γ en la función PWL de la ecuación (2.3), la cual describe las características $I-V$ del diodo de Chua. Por consiguiente, se puede redefinir la ecuación (2.3) como (2.7) [22].

$$i_{NR} = \begin{cases} -g2V_{C1} + (g1-g2)BP_1\gamma, & V_{C1} < -BP_1\gamma \\ -g1V_{C1}, & -BP_1\gamma \leq V_{C1} \leq BP_1\gamma \\ -g2V_{C1} + (g2-g1)BP_1\gamma, & V_{C1} > BP_1\gamma \end{cases} \tag{2.7}$$

Tal como lo ilustra la Figura 2.3(a), la amplitud de la señal $V_{C1}(t)$ es muy pequeña, por lo tanto se añade el parámetro $\gamma = 17$ en la ecuación (2.7). Como resultado se obtienen valores cercanos a 5 y -5 en la señal $V_{C1}(t)$ (Figura 2.3(b)), la cual se va a utilizar como base en el diseño de la técnica de la generación de rutas. Es importante señalar que el escalamiento en amplitud no afecta el comportamiento caótico del sistema.



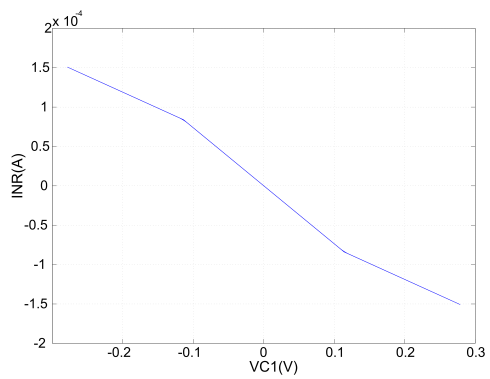
(a) Señal $VC1(t)$ sin escalamiento.

(b) Señal $VC1(t)$ escalada por un factor $\gamma = 17$.

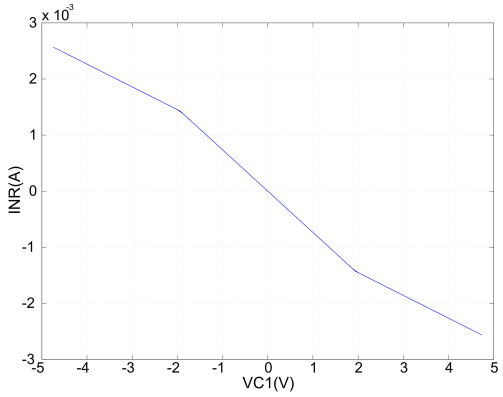
Figura 2.3: Evolución temporal de la señal $VC1(t)$ del circuito de Chua.

2.1.4. Simulación del sistema basado en el circuito de Chua

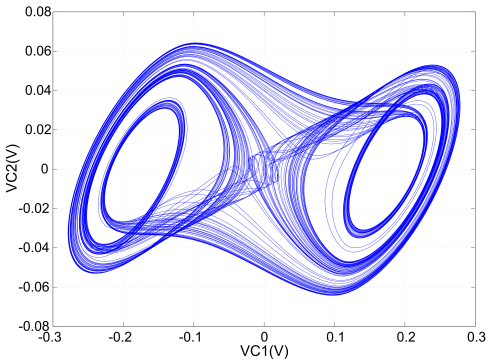
Nos interesa la solución del sistema basado en el circuito de Chua mediante simulaciones numéricas aplicando aproximaciones con el método Forward Euler (FE). Para ello, usamos las ecuaciones (2.4)-(2.6) y consideramos los valores en el circuito de Chua como en [22]: $C1 = 450pF$, $C2 = 1.5nF$, $L = 1mH$, $g1 = 1/1358$, $g2 = 1/2464$, $g3 = 1/1600$, $BP1 = 0.114V$, $BP2 = 0.4V$ y $R = 1625\Omega$. Además, se utiliza un ancho de paso de integración $h = 1 \times 10^{-7}$ y las condiciones iniciales $V_{C1}(0) = 0.01$, $V_{C2}(0) = 0$, $I_L(0) = 0$.



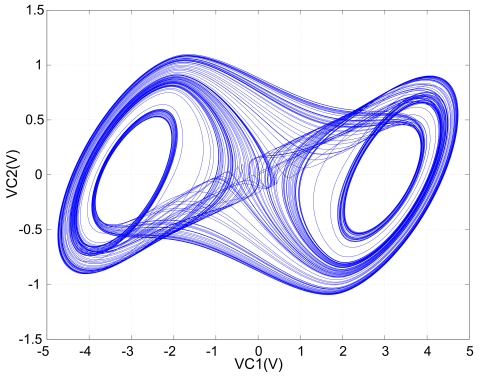
(a) Función PWL sin escalamiento.



(b) Función PWL escalada por un factor $\gamma = 17$.



(c) Atractor sin escalamiento.



(d) Atractor escalado por un factor $\gamma = 17$.

Figura 2.4: Sistema caótico basado en el circuito de Chua.

En la Figura 2.4(a) y 2.4(b) se muestran la gráficas de la función PWL del circuito de Chua antes y después del escalamiento, respectivamente. Del mismo modo en la Figura 2.4(c) y 2.4(d) se muestra el atractor del sistema antes y después del escalamiento.

2.2. Sistema de Chua normalizado

La mayoría de los estudios analíticos sobre el circuito de Chua se han enfocado en ecuaciones adimensionales (normalizadas). En la ecuación (2.8) se muestra la generalización del circuito de Chua. La función $h(x)$ describe una función PWL dada por (2.9), donde q es un número natural ajustado para generar enrollamientos pares o impares, m_0 y m_1 son las pendientes de la función PWL y c son los puntos de quiebre (Figura 2.5.

$$\begin{aligned} \dot{x} &= \alpha[y - h(x)], \\ \dot{y} &= x - y + z, \\ \dot{z} &= -\beta y. \end{aligned} \tag{2.8}$$

$$h(x) \quad = \quad (m_{2q-1})x + \frac{1}{2}\sum_{i=1}^{2q-1}(m_{i-1} - m_i) \times (|x + c_i| - |x - c_i|) \tag{2.9}$$

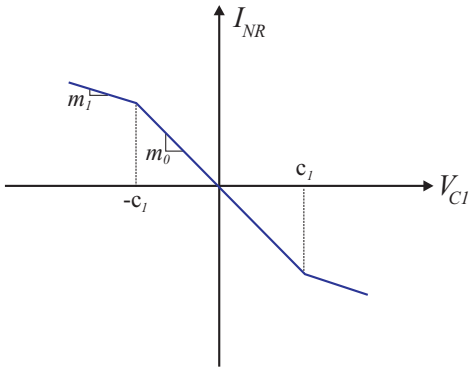
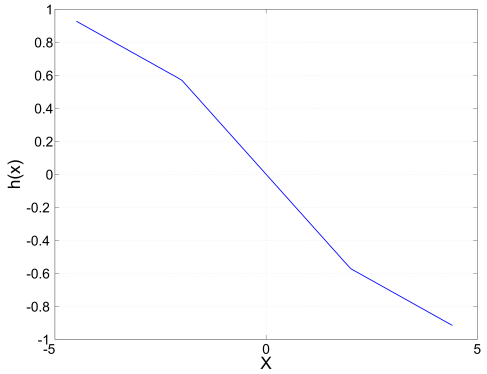


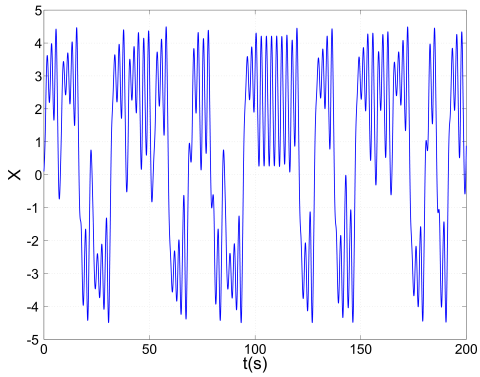
Figura 2.5: Función PWL del sistema de Chua normalizado.

2.2.1. Simulación del sistema de Chua normalizado

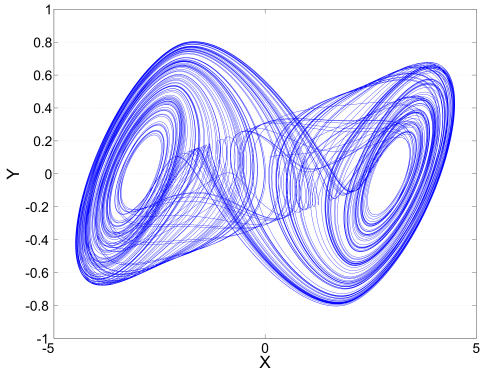
Definiendo los parámetros $\alpha = 9$ y $\beta = 14.286$ del sistema de ecuaciones (2.8) y $q = 1$, $c = 2$, $m_0 = 1/7$ y $m_1 = 2/7$ en la ecuación (2.9), se construye la función PWL $h(x)$ mostrada en la Figura 2.6(a).



(a) Función PWL.



(b) Variable $x(t)$.



(c) Atractor del sistema.

Figura 2.6: Sistema de Chua normalizado.

La Figura 2.6(b) muestra la señal $x(t)$ evolucionando en el tiempo, en la cual podemos observar que definiendo el punto de quiebre $c = 2$ en la función $h(x)$, se obtienen los niveles de excursión adecuados para obtener valores que se encuentren en el rango $[-5 \leq x \leq 5]$ en la amplitud de la señal $x(t)$. Finalmente, la Figura 2.6(c) muestra el atractor producto de graficar las variables de estado x y y de este sistema.

2.3. Sistema de Lorenz

El sistema de Lorenz es un modelo matemático que describe un fenómeno meteorológico conocido como la convección de Rayleigh-Benard y lo reduce a un conjunto de tres ecuaciones diferenciales ordinarias [21]:

$$\begin{aligned} \dot{x} &= \sigma(y - x), \\ \dot{y} &= \gamma x - y - xz, \\ \dot{z} &= xy - bz. \end{aligned} \tag{2.10}$$

Las tres variables físicamente están vinculadas a la intensidad de la convección x, la diferencia de temperatura entre la corriente ascendente y descendente y y la desviación de la temperatura desde el perfil lineal z. Las constantes σ, γ y b son parámetros adimensionales positivos vinculados al problema físico: σ es el número de Prandtl, el cual mide la relación entre la viscosidad del fluido y la difusividad térmica; γ puede ser considerado como la diferencia de temperatura impuesta normalizada (más precisamente, es la relación entre el valor del número de Rayleigh y su valor crítico) y es el principal parámetro de control; finalmente, b es un factor geométrico [11, 21].

2.3.1. Puntos de equilibrio del sistema de Lorenz

Para obtener los puntos de equilibrio es necesario resolver el sistema de ecuaciones:

$$\sigma(y–x) = 0, \tag{2.11}$$

$$\gamma x - xz–y = 0, \tag{2.12}$$

$$xy–bz = 0. \tag{2.13}$$

De la ecuación (2.11) se deduce que $y = x$. Si se sustituye y por x en las ecuaciones (2.12), (2.13) se obtienen las siguientes ecuaciones:

$$x(\gamma - 1 - z) = 0, \tag{2.14}$$

$$-bz + x^2 = 0. \tag{2.15}$$

Una manera de satisfacer la ecuación (2.14) es con $x = 0$, por lo que $z = 0$ en la ecuación (2.15), y como $y = x$ entonces $y = 0$. También es posible satisfacer (2.14) si se elige a $z = \gamma - 1$, entonces (2.15) requiere que $x = \pm\sqrt{b(\gamma - 1)}$ y en consecuencia $y = \pm\sqrt{b(\gamma - 1)}$. Así que los puntos de equilibrio para este sistema quedan de la siguiente manera:

$$\begin{aligned} P_1 &= (0, 0, 0), \\ P_2 &= (\sqrt{b(\gamma - 1)}, \sqrt{b(\gamma - 1)}, \gamma - 1), \\ P_3 &= (-\sqrt{b(\gamma - 1)}, -\sqrt{b(\gamma - 1)}, \gamma - 1). \end{aligned}$$

2.3.2. Escalamiento del sistema de Lorenz

De igual manera que en el sistema basado en el circuito de Chua, es necesario escalar los niveles de excursión (ELs) de las señales del sistema de Lorenz, ya que los valores nominales producen ELs muy grandes. Por consiguiente, es necesario reducir la amplitud de las señales añadiendo los parámetros k_1, k_2 y k_3 al sistema de ecuaciones (2.10), dando como resultado la siguiente transformación del sistema:

$$\begin{aligned} u = \frac{x}{k_1} &\longrightarrow x = uk_1, \\ v = \frac{y}{k_2} &\longrightarrow y = vk_2, \\ w = \frac{z}{k_3} &\longrightarrow z = wk_3. \end{aligned}$$

Así,

$$\begin{aligned} k_1\dot{u} &= \sigma(vk_2 - uk_1), \\ k_2\dot{v} &= \gamma uk_1 - vk_2 - uwk_1k_3, \\ k_3\dot{w} &= uv_1k_2 - bwk_3. \end{aligned}$$

Por lo tanto el sistema queda definido de la siguiente manera:

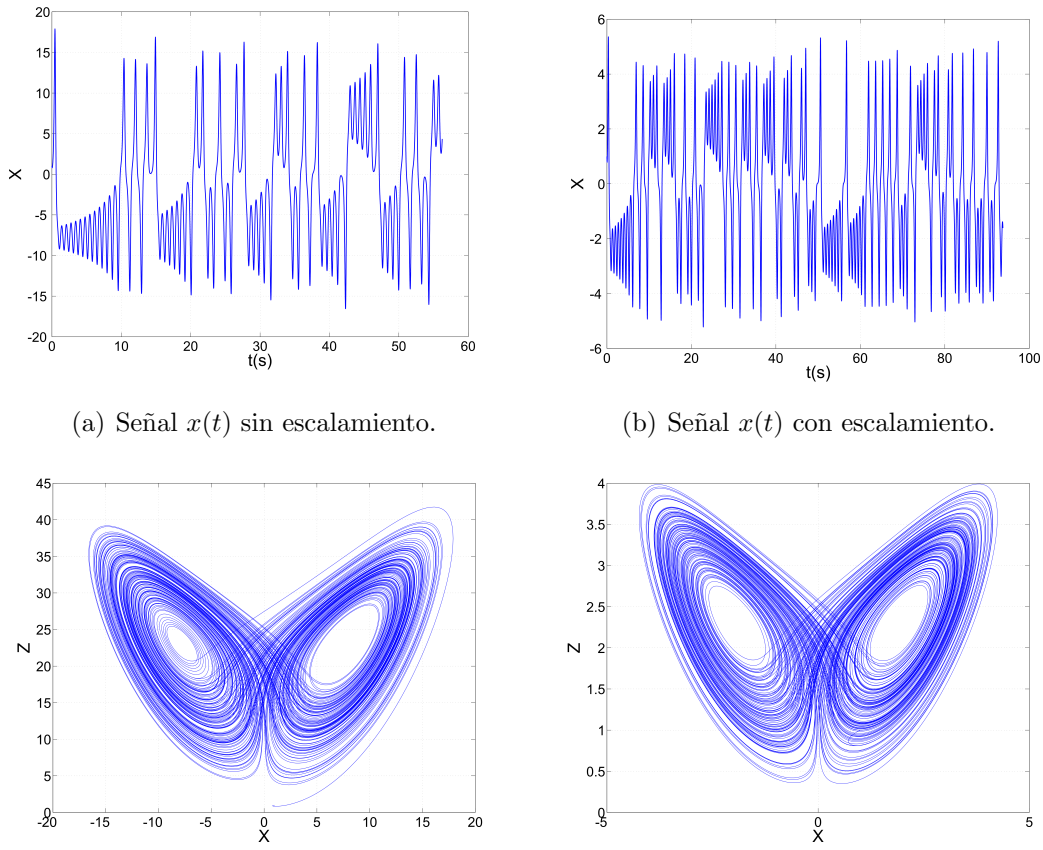
$$\begin{aligned} \dot{u} &= \sigma(v\frac{k_2}{k_1} - u), \\ \dot{v} &= \gamma u\frac{k_1}{k_2} - v - uw\frac{k_1k_3}{k_2}, \\ \dot{w} &= uv\frac{k_1k_2}{k_3} - bw. \end{aligned} \tag{2.16}$$

Además, a consecuencia del escalamiento, los puntos de equilibrio quedarán ubicados en:

$$\begin{aligned} P_1 &= (0,0,0), \\ P_2 &= (\frac{\sqrt{b(\gamma-1)}}{k_1}, \frac{\sqrt{b(\gamma-1)}}{k_2}, \frac{\gamma-1}{k_3}), \\ P_3 &= (-\frac{\sqrt{b(\gamma-1)}}{k_1}, -\frac{\sqrt{b(\gamma-1)}}{k_2}, \frac{\gamma-1}{k_3}). \end{aligned}$$

2.3.3. Simulación del sistema de Lorenz

Ahora bien, definiendo los parametros $\sigma = 10$, $\gamma = 24$ y $b = 8/3$, así como los valores de las constantes de escalamiento $k_1 = 4$ $k_2 = 10$ $k_3 = 10$ en el sistema de ecuaciones (2.16), se obtienen como resultado valores cercanos a 5 y -5 en la señal $x(t)$. La Figura 2.7 muestra la señal $x(t)$ y el atractor del sistema de Lorenz antes y después del escalamiento, como se puede observar el escalamiento no afecta el comportamiento caótico del sistema.



(a) Señal $x(t)$ sin escalamiento.

(b) Señal $x(t)$ con escalamiento.

(c) Atractor sin escalamiento.

(d) Atractor con escalamiento.

Figura 2.7: Señal $x(t)$ y atractor del sistema de Lorenz.

2.4. Sistema basado en una función saturada

El sistema caótico basado en una función de series saturadas se encuentra descrito por la ecuación (2.17), en la que x, y, z son variables de estado, a, b, c y d son constantes positivas, y donde la función $f(x)$ se encuentra definida por la ecuación (2.18). De este modo, en la Figura 2.8 se muestran las características de la serie de funciones saturadas, donde $\pm k$ son llamadas regiones o niveles saturados, $\pm sp$ son los puntos de quiebre y (k/sp) es la pendiente de saturación.

$$\begin{aligned} \dot{x} &= y, \\ \dot{y} &= z, \\ \dot{z} &= -ax - by - cz + df(x). \end{aligned} \tag{2.17}$$

$$f(x) = \begin{cases} k & x > sp \\ (k/sp)x & -sp \leq x \leq sp \\ -k, & x < -sp. \end{cases} \tag{2.18}$$

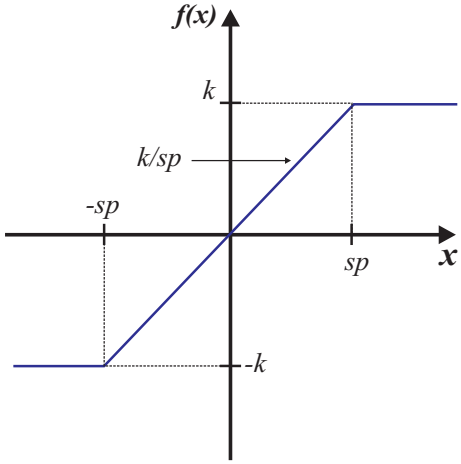


Figura 2.8: Función saturada $f(x)$.

2.4.1. Puntos de equilibrio del sistema basado en una función saturada

Los puntos de equilibrio, como se mencionó anteriormente, se obtienen resolviendo cada una de las ecuaciones del sistema (2.17):

$$\begin{aligned} y &= 0, \\ z &= 0, \\ -ax - by - cz + df(x) &= 0. \end{aligned}$$

En las primeras dos ecuaciones se deduce que $x = y = 0$, así que, sustituyendo x y y en la tercera ecuación se tiene:

$$\begin{aligned} -ax + df(x) &= 0, \\ \frac{df(x)}{a} &= x. \end{aligned}$$

Además, debido a que $a = b = c = d$, se puede definir una $\beta = \frac{d}{a} = 1$, reduciéndose a lo siguiente:

$$(x - f(x)) = 0.$$

Finalmente, evaluando $f(x)$ dada por las ecuaciones (2.18), se obtiene:

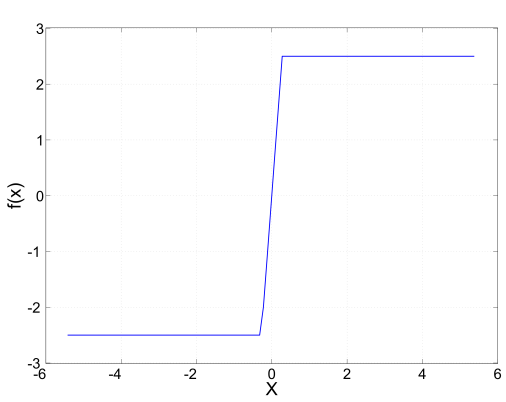
$$x = k,\ x = 0,\ x = -k.$$

De allí pues que los puntos de equilibrio quedan definidos de la siguiente manera:

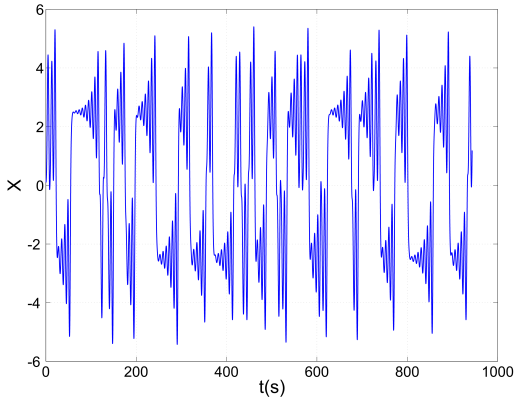
$$\begin{aligned} P1 &= (k, 0, 0), \\ P2 &= (0, 0, 0), \\ P3 &= (-k, 0, 0). \end{aligned}$$

2.4.2. Simulación del sistema caótico basado en una función saturada

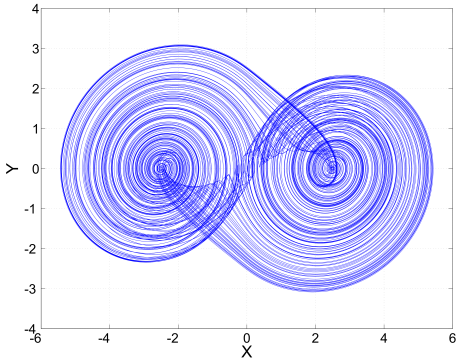
Definiendo los valores de los niveles saturados y el valor de la pendiente es posible construir una función PWL para obtener la amplitud deseada en la señal $x(t)$. Ahora bien, si se define una $f(x)$, dada en (2.18), con los niveles de saturación $\pm k = 2.5$ y una pendiente de $(k/sp) = 100$, se obtiene la función PWL mostrada en la Figura 2.9(a). Finalmente, si se fijan los parámetros $a = b = c = d = 0.7$ en la ecuación (2.17) se obtiene la señal $x(t)$ mostrada en la Figura 2.9(b). Al graficar en el espacio de fase, se forma el atractor mostrado en la Figura 2.9(c), en donde se observa la formación de dos enrollamientos en torno a los puntos de equilibrio del sistema.



(a) Función PWL.



(b) Variable $x(t)$.



(c) Atractor del sistema.

Figura 2.9: Sistema caótico basado en una función saturada.

2.4.3. Oscilador caótico de múltiples enrollamientos

En un sistema basado en series de funciones saturadas el número de enrollamientos que genera el atractor caótico se encuentra dado en función de las regiones saturadas. La ecuación (2.19) describe una aproximación PWL conocida como serie de una función saturada, la cual se muestra en la Figura 2.10, donde $k/\alpha > 0$ es la pendiente, $h > 2$ es el retardo saturado

y p, q son enteros positivos.

$$f(x;k,h,p,q) = \sum_{i=-p}^{q} f_i\,(x;k,h)\,. \tag{2.19}$$

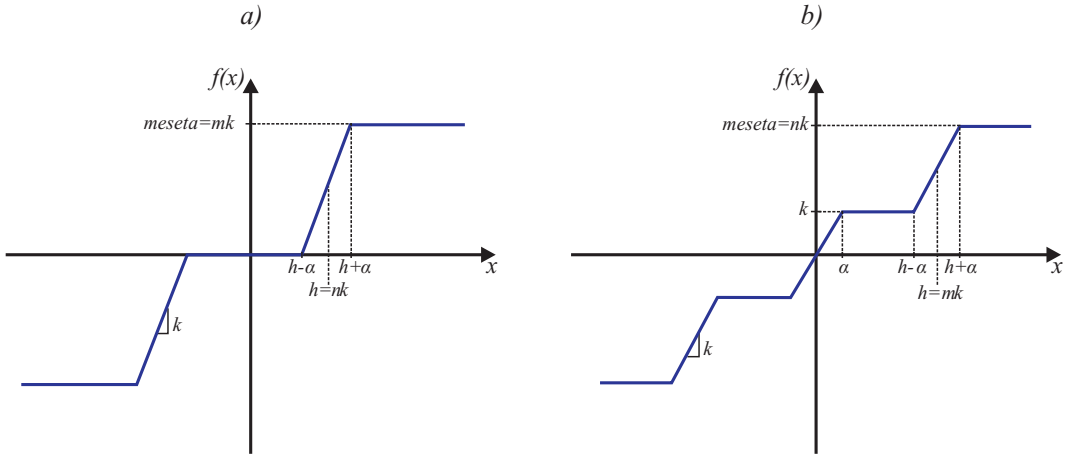


Figura 2.10: Descripción PWL de una serie de funciones saturadas para generar a) enrollamientos impares y b) enrollamientos pares.

De esta manera, la ecuación (2.19) puede reescribirse en una forma explícita como:

$$f(x;\alpha,k,h,p,q) = \begin{cases} (2q+1)k & x > qh+\alpha \\ k/\alpha(x-ih)+2ik & |x-ih| \le \alpha, -p \le i \le q \\ (2i+1)k & ih+\alpha < x < (i+1)h-\alpha, -p \le i \le q-1 \\ -(2p+1)k & x < -ph-\alpha \end{cases} \tag{2.20}$$

Así, un sistema basado en series de funciones saturadas tendrá un atractor caótico de múltiples enrollamientos como se muestra en la siguiente sección para el caso de 1D.

2.4.4. Atractores caóticos de múltiples enrollamientos en 1D

Para generar múltiples enrollamientos orientados en una sola dimensión (1D) se añade una función $f(x;k,h,p,q)$ tal como se muestra en el sistema de ecuaciones (2.21), la cual se encuentra definida por la función (2.20).

$$\begin{aligned} \dot{x} &= y, \\ \dot{y} &= z, \\ \dot{z} &= -ax - by - cz + df(x;k,h,p,q). \end{aligned} \tag{2.21}$$

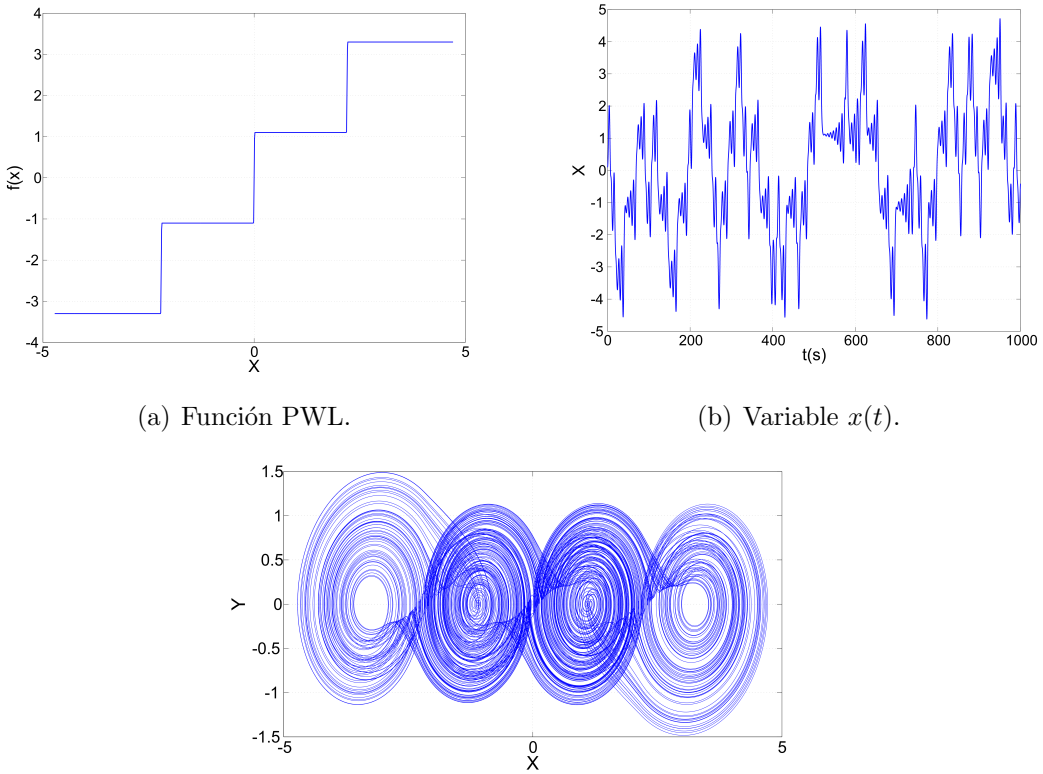
En el sistema de ecuaciones (2.21) existen $2(p+q)+3$ puntos de equilibrio en el eje x, llamados puntos silla de índices 1 y 2, los cuales se definen de acuerdo a la naturaleza de los eigenvalores del sistema. Los enrollamientos se generan solo en torno a los puntos de silla de índice 2, el sistema de ecuaciones (2.21) es capaz de generar $(p+q+2)$ enrollamientos definiendo los valores adecuados en a, b, c, d, k y h. Los responsables de conectar los enrollamientos son los $(p+q+1)$

puntos de silla de índice 1, formando así el atractor. Cabe mencionar que en los puntos de silla de índice 2 un retardo saturado corresponde únicamente a un enrollamiento, mientras que en los puntos de silla de índice 1 una pendiente saturada corresponde a una conexión entre dos enrollamientos vecinos.

La meseta saturada en una función de series saturadas definida por la ecuación (2.20) es: $meseta = \pm nk$ para enrollamientos pares y $meseta = \pm mk$ para enrollamientos impares. Los retardos saturados para los centros de las pendientes están definidas por $h_i = \pm mk$ para enrollamientos pares y $h_i = \pm nk$ para enrollamientos impares, como se muestra en la Figura 2.10. Los factores multiplicativos para las expresiones anteriores son definidas por $n = 1, 3, \cdots, (p+q+1)$ para enrollamientos pares y $n = 1, 3, \cdots, (p+q-1)$ para enrollamientos impares; y $m = 2, 4, \cdots, (p+q)$ para los dos tipos de enrollamientos.

2.5. Simulación del sistema caótico basado en una función saturada con 4 enrollamientos en 1D

En primer lugar se construye la función PWL con el número de pendientes y niveles de saturación necesarios. De acuerdo con la sección anterior, el número de niveles saturados definen el número de enrollamientos, mientras que las pendientes conectan cada uno de los enrollamientos vecinos.



(a) Función PWL.

(b) Variable $x(t)$.

(c) Atractor del sistema.

Figura 2.11: Sistema caótico basado en una función saturada de cuatro enrollamientos.

Después se define el primer nivel de saturación, en este caso $k = 1.1$, el punto de quiebre $\alpha = 0.011$, una pendiente $k/\alpha = 100$, $p = 1$, $q = 1$ y $h = 2.2$, formando así la función PWL $f(x)$ descrita por la ecuación (2.22) y mostrada en la Figura 2.11(a). Finalmente se fijan los parámetros del sistema (2.20) $a = b = c = d = 0.7$ y se integra el sistema mediante simulaciones numéricas en MATLAB. En la Figura 2.11(b) se puede observar que la amplitud de los niveles de excursión de la señal correspondiente a la variable $x(t)$ se encuentra en el rango $[-5 \leq x \leq 5]$. Como consecuencia se obtiene un atractor de cuatro enrollamientos mostrado en la Figura 2.11(c).

$$f(x) = \begin{cases} 3.3 & x > 2.211 \\ 100(x - 2.2) + 2.2 & 2.189 \leq x < 2.211 \\ 1.1 & 0.011 \leq x < 2.189 \\ 100x & -0.011 \leq x < 0.011 \\ -1.1 & -2.189 \leq x < -0.011 \\ 100(x + 2.2) - 2.2 & -2.211 \leq x < -2.189 \\ -3.3 & x < -2.211 \end{cases} \tag{2.22}$$

2.6. Atractores caóticos de múltiples enrollamientos en 2D

Para generar un comportamiento caótico en dos dimensiones (2D) es necesario modificar el sistema caótico dado por la ecuación (2.17). El sistema caótico en 2D se modela aplicando aproximaciones mediante variables de estado como se muestra en la ecuación (2.23), donde x, y, z son variables de estado; a, b, c, d_1 y d_2 son constantes reales positivas.

Para un sistema caótico en dos dimensiones se necesitan dos series de funciones saturadas $f(x)$ y $f(y)$ en el sistema de ecuaciones (2.23) definidas también por la función (2.20), donde p_1, p_2, q_1 y q_2 son enteros positivos. Entonces, el sistema caótico puede crear $(p_1+q_1+2)\times(p_2+q_2+2)$ enrollamientos pares 2D o $(p_1 + q_1 + 1) \times (p_2 + q_2 + 1)$ enrollamientos impares 2D estableciendo adecuadamente los parámetros a, b, c, d_1, d_2, k_1, k_2, h_1 y h_2. Además, la meseta saturada en la función de series saturadas descritas en la ecuación (2.20) es: $meseta = \pm nk$ para enrollamientos pares 2D y $meseta = \pm mk$ para enrollamientos impares 2D. Los retardos saturados para los centros de las pendientes están definidas por $h_i = \pm mk$ para enrollamientos pares 2D y $h_i = \pm nk$ para enrollamientos impares 2D, como se muestra en la Figura 2.10.

Los factores multiplicativos para las expresiones anteriores son definidos por $n = 1, 3, \cdots, (p_2+ q_2 + 1)$ para enrollamientos pares 2D y $n = 1, 3, \cdots, (p_2 + q_2 - 1)$ para enrollamientos impares 2D; $m = 2, 4, \cdots, (p_1 + q_1)$ para los dos tipos de enrollamientos [22].

$$\begin{aligned} \dot{x} &= y - \frac{d_2}{b} f(y; k_2, h_2, p_2, q_2), \\ \dot{y} &= z, \\ \dot{z} &= -ax - by - cz + d_1 f(x; k_1, h_1, p_1, q_1) + d_2 f(y; k_2, h_2, p_2, q_2). \end{aligned} \tag{2.23}$$

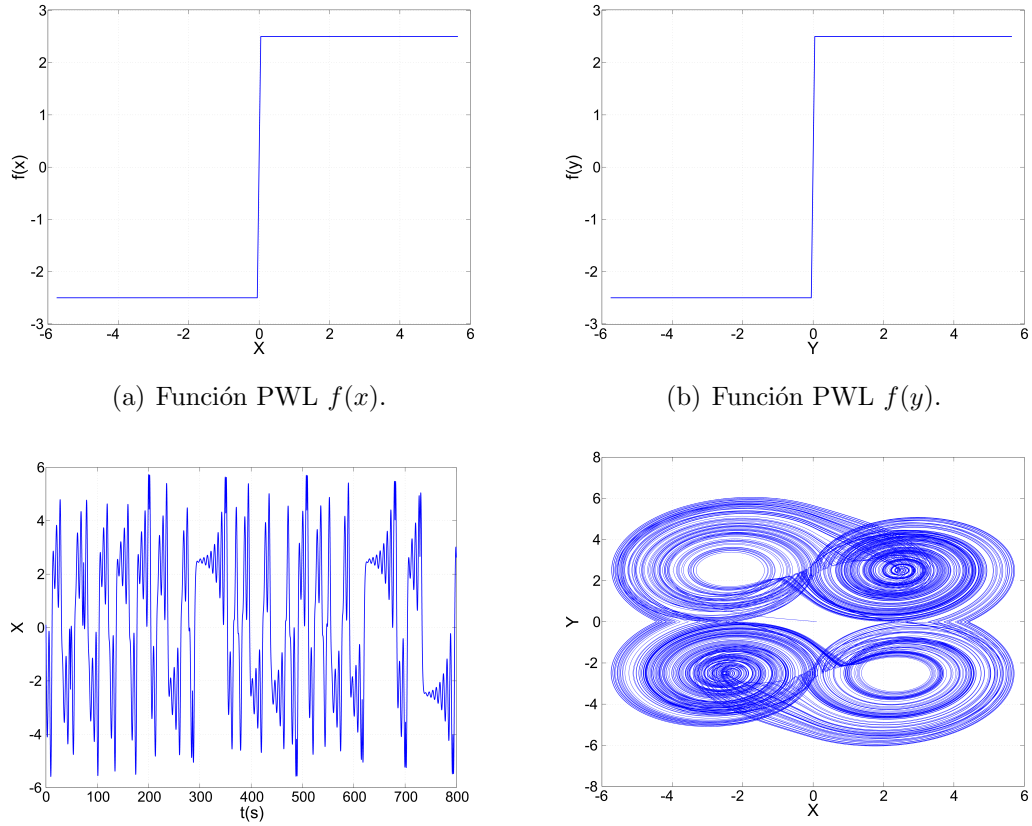
2.6.1. Simulación del sistema caótico basado en una serie de funciones saturadas 2x2

Para realizar la simulación del sistema caótico basado en una serie de funciones saturadas 2x2 es necesario construir dos funciones PWL: $f(x)$ y $f(y)$ con dos niveles de saturación. Después, se deben establecer los parámetros del sistema de ecuaciones (2.23), definiendo $k_1 = k_2 = 2.5$, $pendiente = 100$, $p_1 = p_2 = 1$, $q_1 = q2 = 1$, $h_1 = h_2 = 5$, se forman las funciones $f(x)$ y $f(y)$ dadas por las ecuaciones (2.24) y (2.25), respectivamente.

$$f(x) = \begin{cases} 2.5 & x > 0.025 \\ 100(x) & -0.025 \leq x < 0.025 \\ -2.5 & x < -0.025 \end{cases} \tag{2.24}$$

$$f(y) = \begin{cases} 2.5 & y > 0.025 \\ 100(y) & -0.025 \leq y < 0.025 \\ -2.5 & y < -0.025 \end{cases} \tag{2.25}$$

Fijando los parámetros $a = b = c = 0.7$, $d_1 = d_2 = 0.7$ se resuelve el sistema mediante integración numérica.



(a) Función PWL $f(x)$.

(b) Función PWL $f(y)$.

(c) Señal $x(t)$.

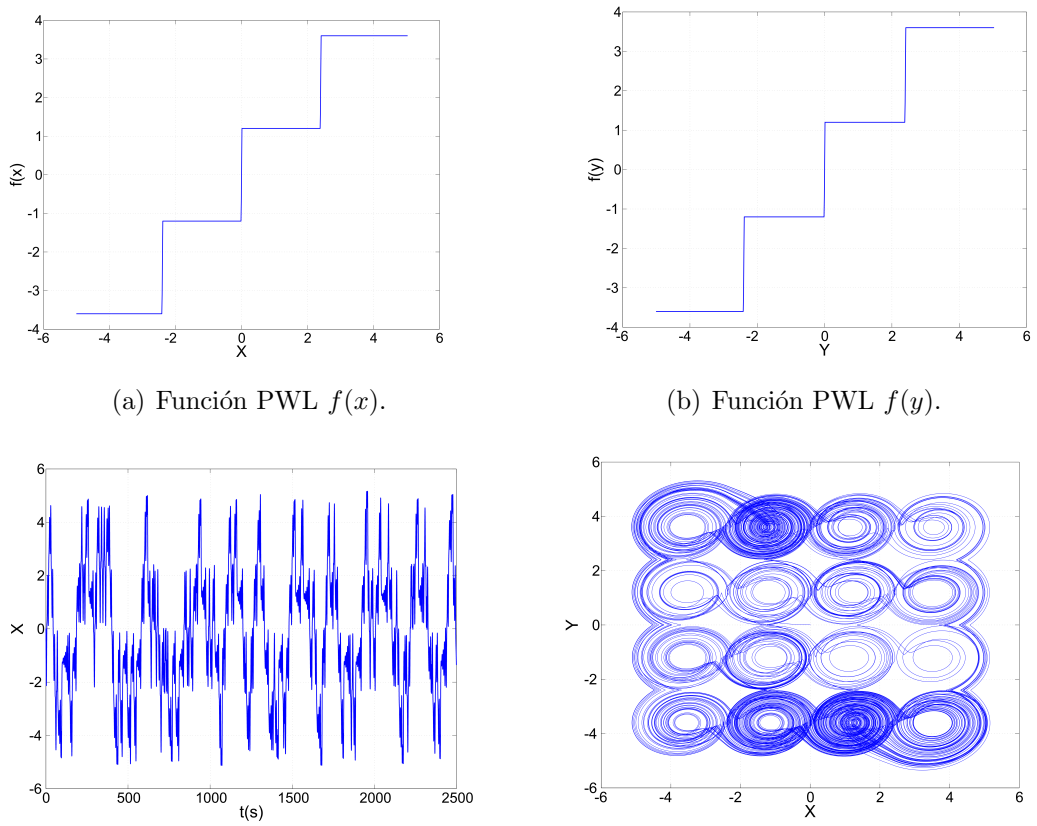
(d) Atractor del sistema.

Figura 2.12: Sistema basado en una serie de funciones saturadas de 2×2.

En la Figura 2.12(a) y (b) se pueden observar la gráficas de las funciones PWL $f(x)$ y $f(y)$, respecticamente, que se aplicarán al sistema; en la Figura 2.12(c) y (d) se observa la señal $x(t)$ y el atractor del sistema, respectivamente.

2.6.2. Simulación de un sistema caótico basado en una serie de funciones saturadas 4x4

Para simular un sistema caótico basado en una serie de funciones saturadas con 4 enrollamientos en dos dimensiones (4x4), es necesario construir dos funciones PWL con cuatro niveles de saturación. Fijando los valores $k_1 = k_2 = 1.2$, *pendiente* $= 100$, $p_1 = p_2 = 1$, $q_1 = q_2 = 1$, $h_1 = h_2 = 2.4$ se forman las funciones $f(x)$ y $f(y)$ mostradas en las ecuaciones (2.26) y (2.27), respectivamente. En la Figura 2.13(a) y (b) se mustran las gráficas de las funciones PWL $f(x)$ y $f(y)$, respectivamente, que se integrarán al sistema de ecuaciones (2.23).



(a) Función PWL $f(x)$.

(b) Función PWL $f(y)$.

(c) Señal $x(t)$.

(d) Atractor del sistema.

Figura 2.13: Sistema basado en una serie de funciones saturadas de 4×4.

Por último, se realiza la integración numérica considerando los valores de los parámetros $a = b = c = 0.7$, $d_1 = d_2 = 0.7$. En la Figura 2.13(c) y (d) se muestra la señal $x(t)$ y el atractor

correspondiente a este sistema, respectivamente.

$$f(x) = \begin{cases} 3.6 & x > 2.412 \\ 100(x - 2.4) + 2.4 & 2.388 \leq x < 2.412 \\ 1.2 & 0.012 \leq x < 2.388 \\ 100x & -0.012 \leq x < 0.012 \\ -1.2 & -2.388 \leq x < -0.012 \\ 100(x + 2.4) - 2.4 & -2.388 \leq x < -2.412 \\ -3.6 & x < -2.412 \end{cases} \tag{2.26}$$

$$f(y) = \begin{cases} 3.6 & y > 2.412 \\ 100(y - 2.4) + 2.4 & 2.388 \leq y < 2.412 \\ 1.2 & 0.012 \leq y < 2.388 \\ 100y & -0.012 \leq y < 0.012 \\ -1.2 & -2.388 \leq y < -0.012 \\ 100(y + 2.4) - 2.4 & -2.388 \leq y < -2.412 \\ -3.6 & y < -2.412 \end{cases} \tag{2.27}$$

Capítulo 3

Caracterización de la dinámica caótica

Este capítulo consta del estudio y el análisis de la dinámica caótica en sistemas dinámicos no lineales mediante simulaciones numéricas en MATLAB. Dicho estudio consta del diseño de dos algoritmos: en el primer algoritmo se presenta un oscilador caótico basado en una serie de funciones saturadas de cuatro enrollamientos, en el cual se analiza la simetría del atractor mediante secciones de Poincaré utilizando un método numérico multipasos de orden variable. En el segundo algoritmo se presenta un mapa de Poincaré del sistema de Lorenz implementando el método de Henon. Los dos algoritmos forman las bases para implementar la técnica en la generación de las rutas de exploración.

3.1. Análisis de la simetría de un atractor caótico de 4x1

El sistema se representa mediante el conjunto de ecuaciones (3.1), donde x, y y z son variables de estado y a, b, c, d son coeficientes reales positivos que se encuentran en el intervalo $0 < a, b, c, d < 1$.

$$\begin{aligned} \dot{x} &= y, \\ \dot{y} &= z, \\ \dot{z} &= -ax - by - cz + df(x; k, h, p, q), \end{aligned} \tag{3.1}$$

La función saturada $f(x; k, h, p, q)$ se encuentra definida por la ecuación (3.2), la cual debe de tener tantos niveles saturados como número de enrollamientos que se deseen generar.

$$f(x; k, h, p, q) = \sum_{i=-p}^{q} f_i\left(x; k, h\right). \tag{3.2}$$

De forma equivalente, la ecuación (3.1) se puede describir en forma matricial como:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a & -b & -c \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ d \end{bmatrix} f\left(x; k, h, p, q\right). \tag{3.3}$$

De igual forma la función PWL (3.2) se puede redefinir en una forma explícita como:

$$f(x;k,h,p,q) = \begin{cases} (2q+1)k & x > qh+1 \\ k(x-ih)+2ik & |x-ih| \leq 1, -p \leq i \leq q \\ (2i+1)k & ih+1 < x < (i+1)h-1, -p \leq i \leq q-1 \\ -(2p+1)k & x < -ph-1 \end{cases} \tag{3.4}$$

3.1.1. Simulación numérica

En el algoritmo se aplican aproximaciones numéricas utilizando el método Forward Euler (FE) descrito por la ecuación (3.5). Se utiliza el método FE para inicializar el método numérico Adams–Moulton (AM) descrito en la ecuación (3.6) para el caso de segundo orden, y por la ecuación (3.7) para el caso de cuarto orden [22, 29].

$$x_{n+1} = x_n + h(f_n), \tag{3.5}$$

$$x_{n+1} = x_n + \frac{h}{2}(f_{n+1} - f_n), \tag{3.6}$$

$$x_{n+1} = x_n + \frac{h}{24}(9f_{n+3} + 19f_{n+2} - 5f_{n+1} + f_n). \tag{3.7}$$

3.1.2. Control Automático del cambio de orden

Un aspecto importante del algoritmo para analizar la simetría de atractor consiste en alternar el orden (segundo y cuarto) durante el proceso de integración, con el objetivo de minimizar los errores de propagación y asegurar la estabilidad del método numérico, así como reducir la complejidad computacional. El cambio de orden se determina por el error de truncamiento, dado por la ecuación (3.8), el cual se evalúa en cada iteración y se compara con un valor de error máximo. Si el error de truncamiento es menor al error máximo el sistema se resuelve con el AM de segundo orden, que consta de un ancho de paso de integración más grande y por tanto reducirá el número de iteraciones. Por otro lado, si el error de truncamiento supera el límite, el sistema se resuelve con el AM de cuarto orden, que posee un ancho de paso de integración más pequeño. Para el método AM de segundo orden se toman los valores $k = 2$ y $C = 1/12$, mientras que para el AM de cuarto orden $k = 4$ y $C = 1/720$.

$$e_{max} = \left[c_k x^{(k+1)}\right] h^{(k+1)}. \tag{3.8}$$

Para calcular los valores del tamaño de paso de integración h con los que se integrará el AM de segundo y cuarto orden es necesario, en primer lugar, obtener los valores propios del sistema, los cuales se obtienen evaluando la ecuación (3.9), donde $\mathbf{A}$ es la matriz de estados, $\mathbf{I}$ es la matriz de identidad y Δ expresa el determinante de $\mathbf{A} - \lambda\mathbf{I}$.

$$\Delta = \det(\mathbf{A} - \lambda\mathbf{I}). \tag{3.9}$$

Una vez obtenidos los eigenvalores, el valor de h se obtiene de la ecuación (3.10), donde λ_{min} es el mínimo absoluto de los eigenvalores expresado por la ecuación (3.11). El valor de Ψ

en la ecuación 3.10 se estima aplicando el teorema de muestreo, por lo tanto, su valor mínimo es 2.

$$h = \frac{\frac{1}{\lambda_{min}}}{\Psi}, \tag{3.10}$$

$$\lambda_{min} = \{|\lambda_1|, \{|\lambda_2|, ..., \{|\lambda_n|\}. \tag{3.11}$$

Se define el oscilador caótico de la función saturada de la ecuación (3.3) con los parámetros $a = b = c = d = 0.7$ y una pendiente de 9, formando así la función (3.12) expresada gráficamente en la Figura 3.1.

$$f(x) = \begin{cases} -27 & x < -19 \\ 9(x+18) - 18 & -19 \leq x < -17 \\ -9 & -17 \leq x < -1 \\ 9x & -1 \leq x < 1 \\ 9 & 1 \leq x < 17 \\ 9(x-18) + 18 & 17 \leq x < 19 \\ 27 & x > 19 \end{cases} \tag{3.12}$$

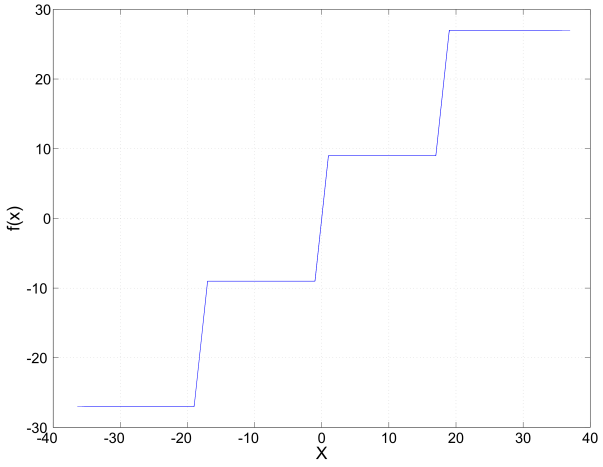


Figura 3.1: Función PWL de cuatro niveles.

Tal como lo ilustra la Figura 3.2, resulta difícil el análisis desde el punto de vista de la evolución temporal de las variables de estado debido a la dinámica tan compleja de los sistemas caóticos. Es por ello que el análisis se efectúa desde el punto de vista del atractor (y contra x) mostrado en la Figura 3.3.

De esta manera, se verifica la simetría del atractor cuantificando las trayectorias que cruzan de un enrollamiento a otro. Para ello se divide la gráfica en tres planos situados en los puntos de equilibrio que unen los enrollamientos, los cuales se ubican en $-18, 0, 18$. Otro análisis consiste en contabilizar las trayectorias que atraviesan un área delimitada cercana al punto de equilibrio en donde se forman los enrollamientos, esto para saber qué tanto se aproxima o se aleja la trayectoria al punto de equilibrio. Para el área delimitada se propuso un círculo de

78.54 $unidades^2$. Los puntos de equilibrio que son los centros de los enrrollamientos se ubican en -27, -9, 9, y 27, como se muestra en la Figura 3.3.

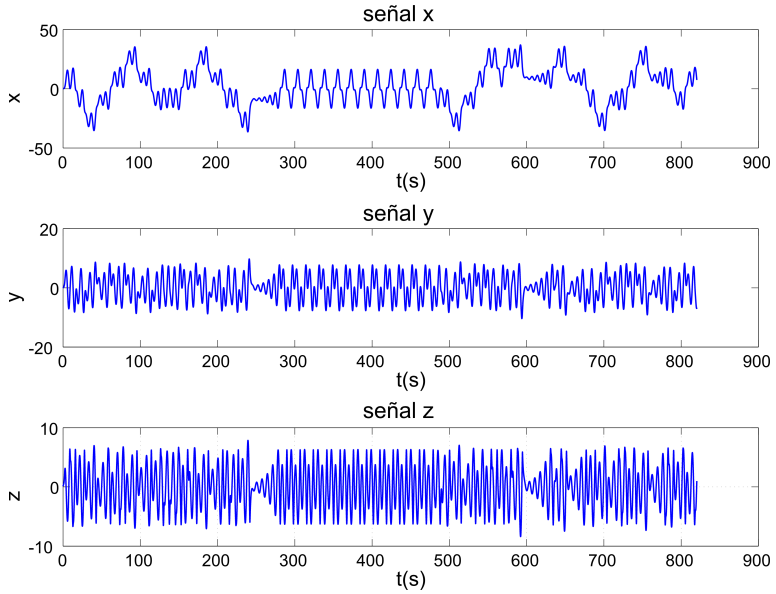


Figura 3.2: Variables $x(t)$, $y(t)$ y $z(t)$ del sistema caótico basado en una serie de funciones saturadas de cuatro enrollamientos.

En la Figura 3.3 se pueden apreciar cuatro circulos rojos, los cuales definen el área delimitada en cada uno de los puntos de equilibrio en donde se forman los enrollamientos. De esta manera es posible determinar cuántas veces son atraídas las trayectorias a cada punto de equilibrio. Asimismo se aprecian los hiperplanos en color rojo, los cuales dividen el plano de fase y cuantifican los cruces de la trayectoria.

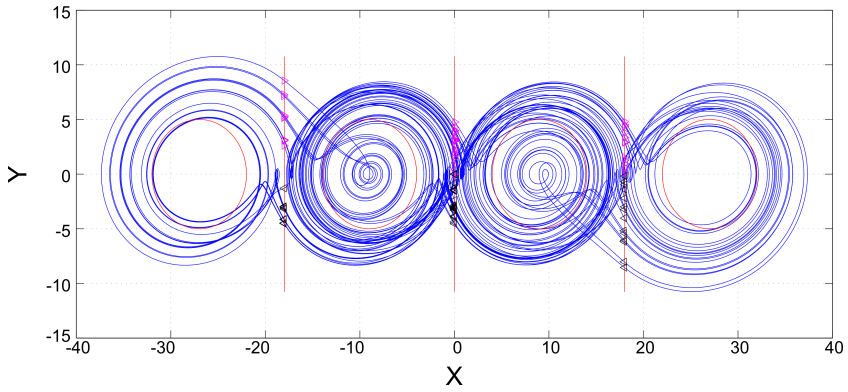


Figura 3.3: Retrato de fase del sistema con los hiperplanos.

En la Tabla 3.1 se presentan los resultados numéricos que describen la simetría del oscilador

caótico.

Variable	**Resultados**
Número total de iteraciones	70000
Número de iteraciones en segundo orden	68199
Número de iteraciones en cuarto orden	1801
Error máximo para cambiar de orden	0.005
Radio de la circunferencia	5
Ancho de paso (2do. orden)	0.01179
Ancho de paso (4to. orden)	0.006862
Puntos de equilibrio	$-27,\ -18,\ -9,\ 0,\ 9,\ 18,\ 27$
Trayectorias que cruzan del plano $1 \to 2$, $2 \to 1$	15
Trayectorias que cruzan del plano $2 \to 3$, $3 \to 2$	7
Trayectorias que cruzan del plano $3 \to 4$, $4 \to 3$	10
Trayectorias que cruzan el área del primer atractor	1
Trayectorias que cruzan el área del segundo atractor	3
Trayectorias que cruzan el área del tercer atractor	8
Trayectorias que cruzan el área del cuarto atractor	2

Cuadro 3.1: Resultados obtenidos del primer algoritmo.

3.2. Análisis mediante el mapa de Poincaré del sistema de Lorenz

El objetivo de este algortimo es analizar mediante el mapa de Poincaré el oscilador caótico del sistema de Lorenz:

$$\begin{aligned} \dot{x} &= \sigma(y - x), \\ \dot{y} &= \gamma x - y - xz, \\ \dot{z} &= xy - bz. \end{aligned} \tag{3.13}$$

La ecuación típica para definir la sección de Poincaré es la siguiente:

$$\begin{aligned} S(x_1, x_2, ..., x_n) &= 0, \\ x_n - a &= 0, \\ x_n &= a. \end{aligned} \tag{3.14}$$

donde si $a = 0$, entonces $x_1 = 0$.

La sección de corte se define en $z = 44.92$, la idea es obtener los valores de x y y en el instante que la señal z cruce la sección. Al graficar en el espacio de fase los valores de x y y en que se cumple el cruce de z, se forma un plano denominado **Mapa de Poincaré**, el cual representa

la evolución de la trayectoria del sistema y comprueba entre otras cosas la no-periodicidad del mismo, la cual es una de las característica que distingue a los sistemas caóticos.

El problema a resolver consiste en obtener el punto exacto $z = 44.92$, esto debido a que la integración numérica se realiza con un ancho de paso determinado, por lo que es muy probable que al momento del cruce de z el punto deseado $z = 44.92$ quede entre x_i y $x_{(i+i)}$, como se ilustra en la Figura 3.4.

Para realizar el ajuste es necesario obtener el valor de t en el que se cumple que $z = 44.92$, para ello es necesario realizar una integración hacia atrás. Cabe considerar que no es una tarea sencilla, pues se desconoce el valor del ancho de paso requerido para integrar de forma inversa y llegar al punto deseado. Es por eso que en esta simulación se hace uso del método de Henon para realizar la integración inversa y así regresar al punto deseado.

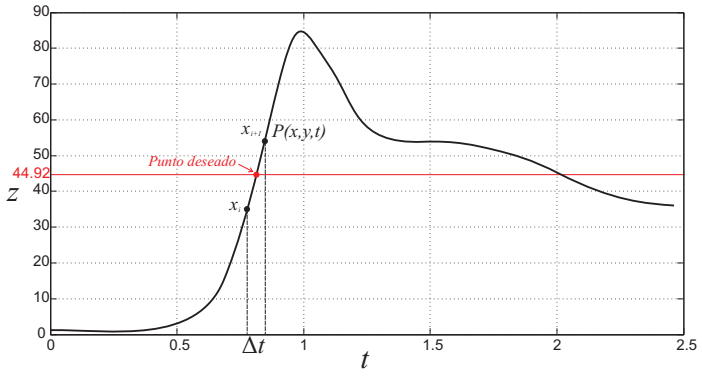


Figura 3.4: Sección de Poincaré establecido en $z = 44.92$

3.2.1. Método de Henón

Los parámetros en el sistema de Lorenz (3.13) se definieron de la siguente manera: $\sigma = 16$, $\gamma = 45.92$ y $b = 24$, el cual tiene la forma:

$$\begin{aligned} \frac{dx_1}{dt} &= f_1(x_1,\ x_2, \ldots,\ x_n), \\ \frac{dx_2}{dt} &= f_2(x_1,\ x_2, \ldots,\ x_n), \\ \frac{dx_n}{dt} &= f_n(x_1,\ x_2, \ldots,\ x_n). \end{aligned} \tag{3.15}$$

Por su parte, el método de Henón, consiste en lo siguiente:

1. Se empieza integrando el sistema con un ancho de paso h fija calculada con la ecuación (3.10).

2. Se detiene la integración cuando la trayectoria $z \geq 44.92$.

3. Se integra de forma inversa el sistema (3.15) mostrado en la ecuación (3.16), con un ancho

de paso de integración $-\Delta z$, donde $\Delta z = x_{(i+1)} - 44.92$.

$$\begin{aligned} \frac{dx_1}{dx_n} &= \frac{f_1}{f_n}, \\ &\vdots \\ \frac{dx_{(n-1)}}{dx_n} &= \frac{f_{(n-1)}}{f_n}, \\ \frac{dt}{dx_n} &= \frac{1}{f_n}. \end{aligned} \tag{3.16}$$

En la Figura 3.5 se observa que el equivalente del sistema inverso es rotar la gráfica noventa grados a la izquierda. Desde esta perspectiva, al integrar el sistema inverso con un ancho de paso $h = -\Delta z$ se obtienen los valores de x, y y t.

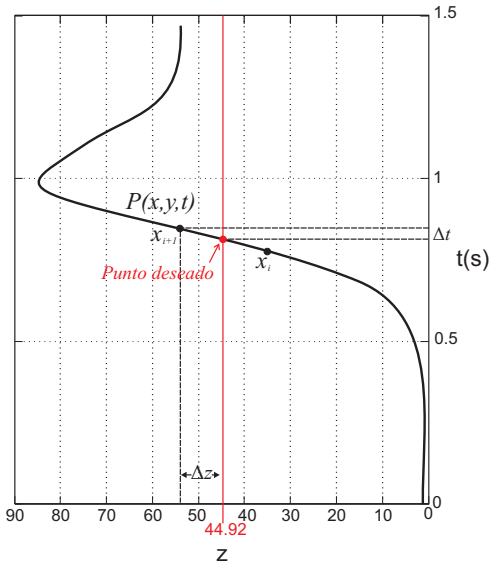


Figura 3.5: Señal z vista desde el sistema inverso.

4. Finalmente se integra nuevamente el sistema de normalmente a partir del punto corregido.

La Figura 3.6 muestra un acercamiento en la señal correspondiente a la variable z, el punto color magenta representa el valor de $x_{(i+1)}$, es decir, el valor de z cuando cruza la sección, en ese momento la integración se detiene y se realiza la integración hacia atrás para obtener los valores de x, y y t.

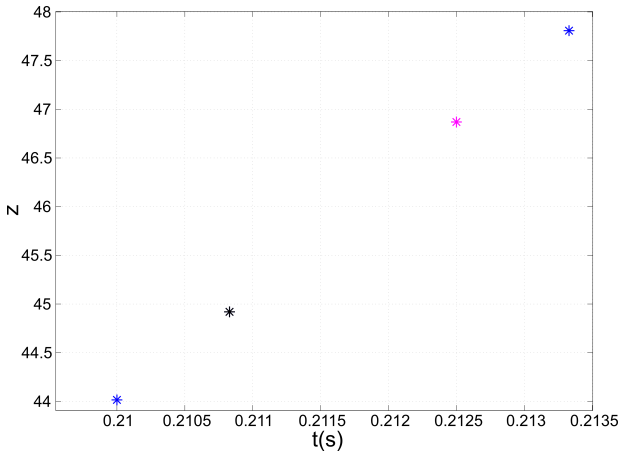


Figura 3.6: Acercamiento de la señal z.

El punto en color negro equivale al punto corregido en el que $z = 44.92$ y a partir de ahí se realiza la integración de forma normal. En la gráfica de la Figura 3.7 se observan la evolución temporal de las variables x y y. Los puntos en color rojo indican el valor de x y de y respectivamente, cuando $z = 44.92$. Finalmente observamos en la Figura 3.8 la gráfica de x

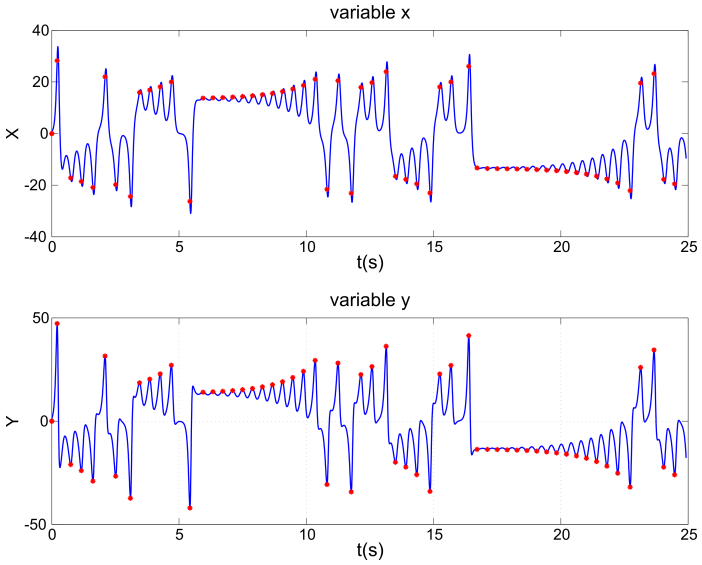


Figura 3.7: Evolución temporal de x y y con el punto en el que $z = 44.92$.

contra y cuando el valor de $z = 44.92$, la cual representa el mapa de Poincaré del sistema de Lorenz.

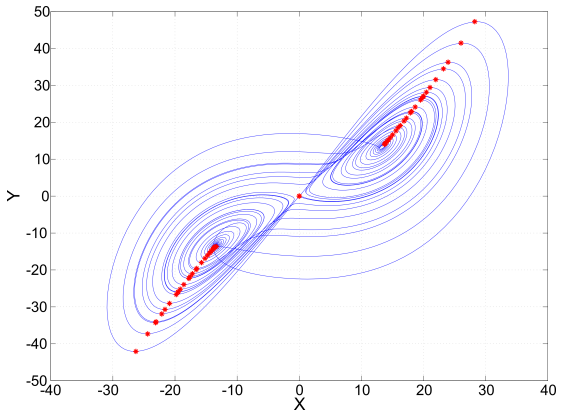


Figura 3.8: Mapa de Poincaré del sistema de Lorenz.

Capítulo 4

Técnicas para generar rutas de exploración

En este capítulo se presenta la técnica empleada para la planeación de rutas de exploración utilizando los sistemas caóticos de Chua, Lorenz y el sistema basado en series de funciónes saturadas. La técnica que se propone para planear las trayectorias de exploración consiste en el diseño de generadores de números aleatorios (RNGs por sus siglas en inglés). Una razón por la que se utilizan RNGs es debido a que es posible validar la eficiencia de estos mediante pruebas estadísticas de aleatoriedad.

4.1. Generadores de números aleatorios

Un generador de números aleatorios es una fuente impredecible de números. Matemáticamente se define como una fuente de largas secuencias de símbolos independientes e identicamente distribuídos [32]. Existen básicamente dos tipos de generadores utilizados para producir secuencias aleatorias: Generadores de números aleatorios (RNGs) y generadores de números pseudoaleatorios (PRNGs).

Los generadores (RNG) utilizan generalmente una fuente no determinista (fuente de entropía) junto con alguna función de procesamiento para producir aleatoriedad. Es necesario contar con un post-procesamiento para superar cualquier debilidad en la fuente de entropía, no hacerlo resulta en la producción de números no aleatorios (por ejemplo, la aparición de cadenas largas de ceros o unos). La fuente de entropía consiste típicamente de alguna cantidad física, como el ruido en un circuito eléctrico, procesos de interrupción por parte del usuario (por ejemplo, pulsaciones de teclas o movimientos del mouse), los efectos cuánticos en un semiconductor o utilizando diversas combinaciones de las entradas antes mencionadas. Las salidas de un generador de números aleatorios se pueden usar directamente como un número aleatorio o pueden ser alimentados en un generador de números pseudoaleatorios (PRNG). Para ser utilizado directamente, la salida de cualquier RNG debe satisfacer criterios estrictos de aleatoriedad medidas por las pruebas estadísticas para determinar que las fuentes físicas de las entradas RNG aparecen al azar [20].

Un PRNG utiliza una o más entradas y genera múltiples números "pseudoaleatorios". A las entradas de un PRNG se llaman semillas, las cuales deben ser tambien aleatorias e impredecibles. 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 RNG como compañero.

Las salidas de un PRNG son funciones normalmente deterministas de la semilla; es decir, toda verdadera aleatoriedad se limita a la generación de las semillas. La naturaleza determinista del proceso conduce a la expresión "pseudo". Dado que cada elemento de una secuencia pseudoaleatoria es reproducible de su semilla, sólo la semilla necesita ser guardada si se requiere la reproducción o la validación de la secuencia pseudoaleatoria.

Irónicamente, los números pseudoaleatorios a menudo parecen poseer más aleatoriedad que los números aleatorios obtenidos de fuentes físicas. Si una secuencia pseudoaleatoria se construye correctamente, cada valor de la secuencia se produce a partir del valor anterior a través de transformaciones que introducen aleatoriedad adicional. Una serie de estas transformaciones puede eliminar auto-correlaciones estadísticas entre la entrada y la salida. Por lo tanto, las salidas de un PRNG pueden tener mejores propiedades estadísticas y ser más rápidos que un generador de números aleatorios.

Una secuencia de bits aleatorios podría interpretarse como el resultado de los lanzamientos de una moneda con lados etiquetados como "0" y "1", con una probabilidad de exactamente la mitad de la obtención de un "0" o "1". Por otra parte, todos los elementos de la secuencia deben ser independientes uno de otro, y el valor del siguiente elemento en la secuencia no debe ser predecible independientemente del número de elementos que ya se hayan producido [20].

4.1.1. Generación de las rutas de exploración a partir de un solo sistema caótico

La técnica para generar números aleatorios que se va a utilizar en este trabajo consiste básicamente de cinco bloques Figura 4.1 [33]:

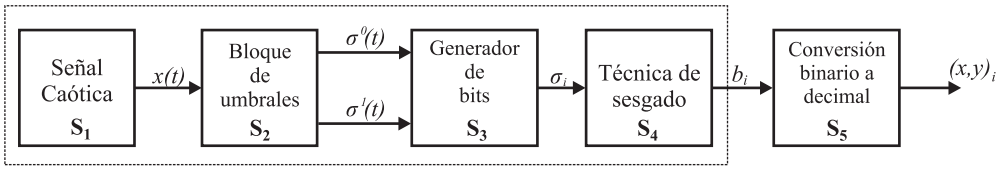
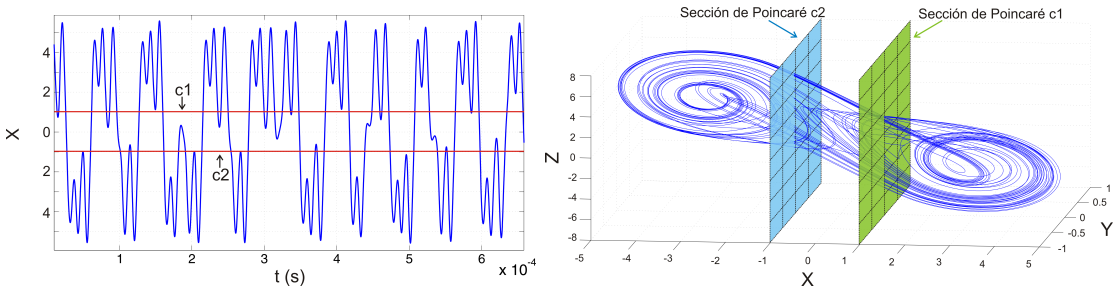


Figura 4.1: Diagrama de bloques general del RNG.

- En el bloque $\mathbf{S_1}$ se analiza la señal temporal $x(t)$ del sistema y se proponen secciones delimitadas por umbrales ($c1$ y $c2$) que la intersecten de forma horizontal como se muestra en la Figura 4.2.
- En el bloque $\mathbf{S_2}$ se obtienen bits cuando la señal temporal atravieza la secciones de corte



(a) Señal caótica $x(t)$ (b) Atractor caótico

Figura 4.2: Proceso de muestreo de la señal caótica $x(t)$, usando los umbrales c_1 y c_2.

definidas. La forma en que se obtienen los bits es la siguiente:

$$S_2 : \begin{array}{l} \sigma^1(x(t)) = \begin{cases} 0, & \text{si } x(t) < c_1 \\ 1, & \text{si } x(t) \geq c_1 \end{cases} \\ \\ \sigma^0(x(t)) = \begin{cases} 0, & \text{si } x(t) > c_2 \\ 1, & \text{si } x(t) \leq c_2 \end{cases} \end{array}$$

- En el bloque $\mathbf{S_3}$ se genera una secuencia de bits dada por la fórmula:

$$S_3 : \sigma_i(\sigma^0, \sigma^1) \quad = \quad \begin{cases} 0, & \text{si } \sigma^0 = 0, \sigma^1 : 0 \uparrow^1 \\ 1, & \text{si } \sigma^1 = 0, \sigma^0 : 0 \uparrow^1 \end{cases}$$

  donde σ^0:$0 \uparrow^1$ muestra la transición lógica de '0' a '1' de σ^0 e $i \in \{0, 1, 2..., n\}$.

- El bloque $\mathbf{S_4}$ implementa la técnica de Von Neumann (VN) [33] para descartar bits repetidos, la cual consiste en convertir el par de bits '01' en la salida '0', '10' en la salida '1' y se descartan los pares '00' y '11'.

- Finalmente en el bloque $\mathbf{S_5}$ se hace una conversión de binario a decimal para construir el par de coordenadas (x,y).

Como resultado, cada generador de números aleatorios diseñado, tiene una serie de coordenadas (x,y), las cuales serán las trayectorias de movimiento del robot móvil.

En el algoritmo es necesario incorporar una sección que cuantifique el porcentaje de cobertura con base en las coordenadas que fueron visitadas, para ello se propone un área de 30x30 unidades, lo cual se traduce en un total de 900 coordenadas posibles. Entonces, para generar un par de coordenadas (x,y) se necesitan un total de 10 bits, 5 bits para x y 5 bits para y, desechando los números 0 y 31.

Debido a la complejidad de la dinámica de los sistemas caóticos resulta difícil deducir los valores óptimos de las secciones de corte $c1$ y $c2$, por lo que es necesario realizar un barrido para determinar las secciones que presenten mejor porcentaje de cobertura. El barrido se realiza en la sección c_1 de 1 a 5 y en la sección c_2 de -1 a -5 con una resolución de 0.1, dando un total de cerca de 1600 combinaciones diferentes.

En cada combinación de c_1 y c_2 se generan 3000 números aleatorios, formando así 1500 coordenadas, es decir, 1500 trayectorias de exploración. Se consideran 1500 trayectorias debido a que la curva del porcentaje de cobertura presenta un comportamiento exponencial, por consiguiente las probabilidades de obtener una coordenada nueva se van reduciendo de forma importante, además se considera que son suficientes para cubrir la mayor parte de la superficie y así determinar si la combinación c_1 y c_2 es eficiente en la generación de trayectorias de exploración.

4.1.2. RNGs combinando dos señales caóticas a diferente frecuencia

Otra técnica propuesta para generar números aleatorios consiste en combinar la dinámica de dos sistemas caóticos, la cual, como se muestra en la Figura 4.3, se consideran básicamente cuatro bloques:

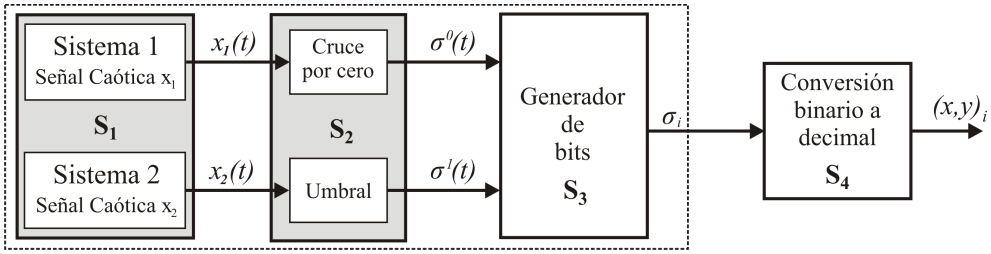


Figura 4.3: Diagrama a bloques del RNG que combina dos sistemas caóticos a diferente frecuencia.

- En el bloque $\mathbf{S_1}$ se analiza una señal temporal lenta $x_1(t)$ y una señal temporal rápida $x_2(t)$, las cuales corresponden a diferentes sistemas caóticos.

- En el bloque $\mathbf{S_2}$ se obtienen bits cuando la señal temporal $x_1(t)$ cruza por cero y cuando la señal $x_2(t)$ se encuentra por debajo o por encima la sección de Poincaré ubicada en el punto de equilibrio que une los enrollamientos en x. La forma en que se obtienen es la siguiente:

$$S_2 : \begin{array}{l} \sigma^0(x_1(t)) = \begin{cases} 0, & \text{si } x_1(t) < 0 \\ 1, & \text{si } x_1(t) \geq 0 \end{cases} \\ \sigma^1(x_2(t)) = \begin{cases} 0, & \text{si } x_2(t) > 0 \\ 1, & \text{si } x_2(t) \leq 0 \end{cases} \end{array}$$

- En el bloque $\mathbf{S_3}$ se genera una secuencia de bits dada por:

$$S_3 : \sigma_i(\sigma^1, \sigma^0) \quad = \quad \begin{cases} 0, & \text{si } \sigma^1 = 0, \sigma^0 : 0 \uparrow^1 \\ 1, & \text{si } \sigma^1 = 1, \sigma^0 : 0 \uparrow^1 \end{cases}$$

  donde σ^0:$0 \uparrow^1$ muestra la transición lógica de '0' a '1' de σ^0 e $i \in \{0, 1, 2..., n\}$.

  En el análisis de los RNGs diseñados con esta técnica también se requerirá de bits generados a partir de un cruce por cero bidireccional, para dicho análisis el bloque $\mathbf{S_3}$ se

modifica de la siguiente manera:

$$S_3 : \sigma_i(\sigma^1, \sigma^0) \quad = \quad \begin{cases} 0, & \text{si } \sigma^1 = 0, (\sigma^0 : 0 \uparrow^1 \vee \sigma^0 : 1 \downarrow_0) \\ 1, & \text{si } \sigma^1 = 1, (\sigma^0 : 0 \uparrow^1 \vee \sigma^0 : 1 \downarrow_0) \end{cases}$$

donde σ^0:1 $\downarrow_0$ muestra la transición lógica de '1' a '0' de σ^0.

- Finalmente en el bloque **S_4** se hace una conversión de binario a decimal para construir el par de coordenadas (x,y).

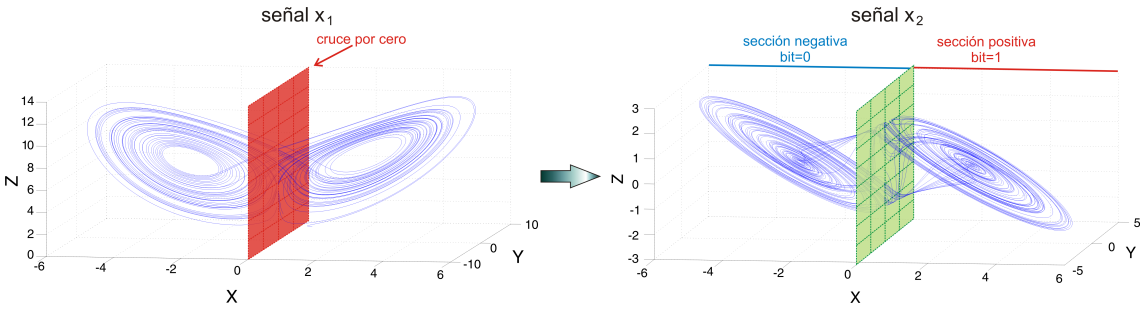


Figura 4.4: Diagrama para combinar dos sistemas caóticos.

En la Figura 4.4 se pueden observar dos atractores correspondientes a diferentes sistemas. Cada vez existe un cruce por en cero en la señal x_1, el programa evalúa si la señal x_2 se encuentra en la parte positiva para generar un "1" o generar un "0" si se encuentra en la parte negativa. El diagrama de flujo de la Figura 4.5 ilustra la la manera en que se obtienen los bits aleatorios con esta técnica.

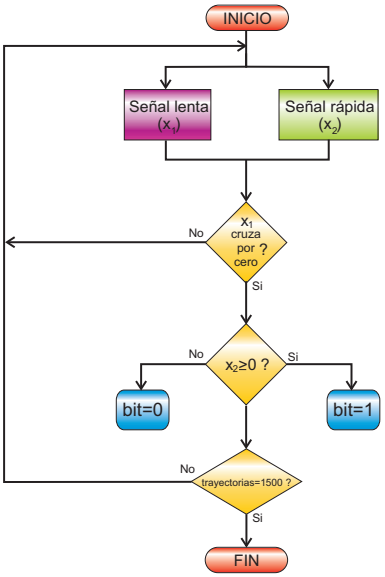


Figura 4.5: Diagrama de flujo del RNG combinando dos sistemas caóticos.

Para llevar a cabo esta técnica en la obtención de bits aleatorios es necesario realizar un análisis para determinar el factor de escalamiento óptimo de la señal rápida en un flanco

de subida y en un flanco bidireccional, esto debido a que en un flanco bidireccional los bits se obtienen dos veces más rápido, sin embargo es necesario determinar qué tanto se compromete el porcentaje de cobertura.

Es importante señalar que en este tipo de RNGs NO SE IMPLEMENTÓ LA TÉCNICA DE VON NEUMANN debido a que, a diferencia de los RNGs diseñados con un solo sistema, se obtuvieron buenos resultados en el porcentaje de cobertura sin necesidad de técnicas de postprocesamiento.

4.1.3. Técnica de generación de trayectorias a partir de sistemas caóticos híbridos

Esta técnica consiste en la generación de rutas a partir de RNGs combinando los sistemás dinámicos no lineales con comportamiento caótico de Chua, Lorenz y el sistema basado en una función saturada. La idea central consiste en la conmutación entre dos sistemas caóticos durante el proceso de integración con el objetivo de crear una dinámica más compleja, y principalmente que las fuentes de entropía no estén correlacionadas. La Figura 4.6 muestra los bloques que conforman este tipo de RNGs.

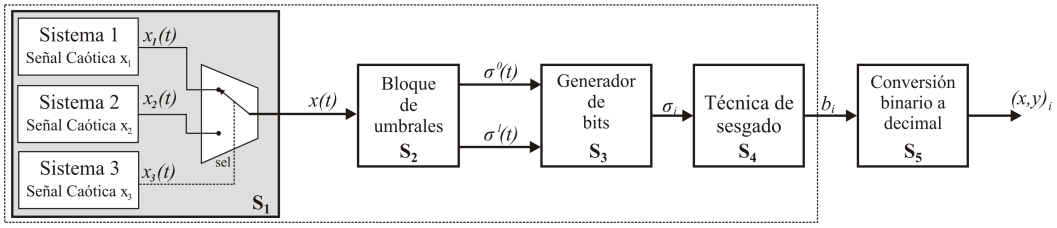
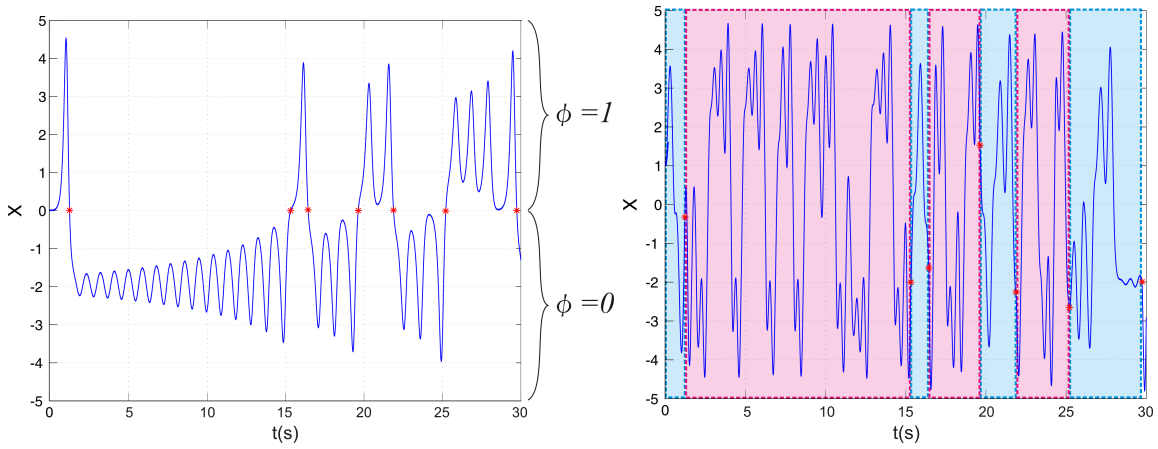


Figura 4.6: Diagrama de bloques que describen los RNGs híbridos.

En este caso en el bloque $\mathbf{S_1}$, se lleva a cabo una conmutación entre dos sistemas caóticos, la cual es determinada mediante los bits generados por la señal $x_3(t)$ correspondiente a un tercer sistema caótico. Esto es, evaluando si la señal $x_3(t)$ se encuentra en su parte positiva o negativa durante el proceso de integración (Figura 4.7 a)). Dichos bits de control se obtienen de la siguiente manera:

$$\phi(x_3(t)) = \begin{cases} 0, & \text{si } x_3(t) < 0 \\ 1, & \text{si } x_3(t) \geq 0 \end{cases}$$

Así, cuando $\phi = 1$ el sistema se integra con el sistema caótico número uno y cuando $\phi = 0$ el sistema se integra con sistema caótico número dos. Como resultado se obtiene una señal $x(t)$ compuesta por dos sistemas caóticos como la que se muestra en la Figura 4.7 b).



(a) Obtención de los bits ϕ. (b) Señal $x(t)$ combinada.

Figura 4.7: Construcción de la señal híbrida $x(t)$.

En los bloques $\mathbf{S_2}$, $\mathbf{S_3}$, $\mathbf{S_4}$ y $\mathbf{S_5}$ se aplica la técnica de obtención de bits descrita en el diagrama de bloques de la Figura 4.1. Finalmente se lleva a cabo el proceso para encontrar las secciones de Poincaré óptimas mediante el barrido de la señal.

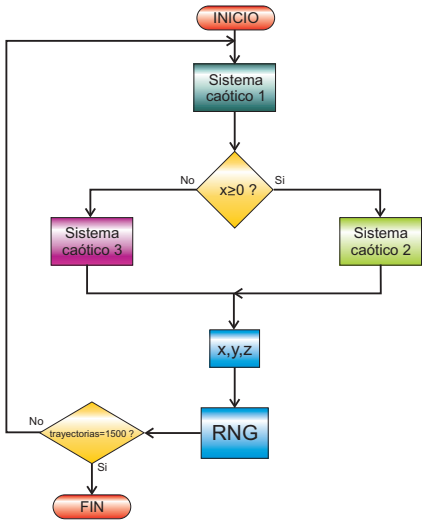


Figura 4.8: Diagrama de flujo de los RNGs híbridos.

La Figura 4.8 muestra un diagrama de flujo generalizado del programa para combinar los sistemas caóticos y formar el RNG híbrido. Cabe mencionar que combinar los sistemas caóticos no es tarea sencilla debido a que al momento de la conmutación entre los sistemas eventualmente se dará el caso en que los valores ya sea de x, y o z quedarán fuera de la region de atracción del sistema en turno, esto es debido a que los puntos de equilibrio no son los mismos en todos los sistemas, lo que conlleva a que la trayectoria del atractor diverja al infinito. Un ejemplo de lo anterior es mostrado en la Figura 4.4, en la cuál se observa el atractor del sistema de Lorenz

y el atractor del sistema basado en una función saturada. Se puede observar que los valores de z en el sistema de Lorenz oscilan únicamente en la parte positiva, mientras que los valores de z en el sistema de la función saturada oscilan en la sección positiva y negativa, esto ocaciona que al momento de la conmutación, la trayectoria del atractor se va alejando cada vez mas de la region de atracción de cada uno de los sistemas hasta divergirá al infinito.

Para solucionar este problema se analizaron los niveles de excursión de las variables x, y y z de los sistemas caóticos que se iban a utilizar y se propuso escalar las señales que se salieran del rango con respecto al sistema que continuará la integración numerica o incluso en el caso particular del sistema de Lorenz donde los valores de z solo toma valores positivos, introducir el valor de $z = |z|$ para acercar las trayectorias a sus puntos de equilibrio. Como resultado se obtiene un atractor combinado como el de la Figura 4.9.

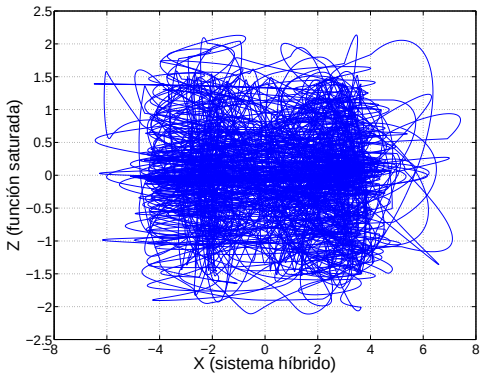


Figura 4.9: Atractor híbrido.

4.1.4. RNGs a partir de sistemas caóticos con exponentes de Lyapunov optimizados

En esta sección se analiza la dependencia del máximo exponente de Lyapunov en la generación de números aleatorios. Para ello se analiza el sistema de Chua normalizado (Mostrado en el Apéndice A) y el sistema basado en una función saturada de 2x1. El análisis consiste en utilizar la técnica de generación de bits aleatorios descrito en la Sección 3.1.1, después de realizar el barrido en la señal para obtener las secciones de Poincaré óptimas se modifican los parámetros de los sistemas para optimizar el máximo exponente de Lyapunov y se repite el proceso. Para cada sistema los parámetros se modifican en dos ocaciones, esto para saber si existe alguna correlación entre el porcentaje de cobertura y el máximo exponente de Lyapunov, el cual se calcula mediante aproximaciones por series de tiempo. La descripción y el cálculo de las pendientes que aproximan los valores del máximo exponente de Lyapunov se encuentran el Apéndice C.

En la Tabla 4.1 se observan los sistemas caóticos que se utilizaron para este análisis así como el máximo exponente de Lyapunov correspondiente a cada valor en los parámetros de los sistemas.

Sistema	h	**Parámetros**	**Max. E.L.**
Chua Norm. (L1)	0.001	$\alpha = 9$, $\beta = 14.286$	0.0002153
Chua Norm. (L2)	0.001	$\alpha = 16.57$, $\beta = 25.88$	0.000273
Chua Norm. (L3)	0.001	$\alpha = 17.96$, $\beta = 31.66$	0.0008638
Saturada 2x1 (L1)	0.0152	$a = 0.7$, $b = 0.7$, $c = 0.7$, $d = 0.7$	0.0011
Saturada 2x1 (L2)	0.0152	$a = 1$, $b = 1$, $c = 0.0499$, $d = 1$	0.0018
Saturada 2x1 (L3)	0.0152	$a = 0.7$, $b = 0.788$, $c = 0.643$, $d = 0.666$	0.0031

Cuadro 4.1: Parámetros para el análisis del máx. Exponente de Lyapunov

Capítulo 5

Diseño de RNGs basados en sistemas caóticos

En este capítulo se describen los RNGs que se diseñaron a partir de los sistemas caóticos de Chua, Lorenz y el sistema basado en una función saturada. En primer lugar se presenta el RNG basado en el circuito de Chua en donde se hace un estudio para comprobar la importancia de las técnicas de post-procesamiento en la generación de números aleatorios, así como los RNGs basados en un solo sistema caótico descritos en el capítulo anterior. En segundo lugar se presentan los RNGs utilizando dos sistemas caóticos a diferente frecuencia, seguido por el RNG híbrido. Finalmente se presentan los RNGs en donde se analiza la dependencia del máximo exponente de Lyapunov en la generación de números aleatorios.

Para analizar cada RNG es necesario obtener datos importantes que nos permitan evaluar la eficiencia de cada uno de ellos. Para ello se analizan cuatro gráficas: la primera gráfica consta de una superficie en donde se muestran los porcentajes de cobertura de todas las secciones de Poincaré incluídas en el barrido. La segunda gráfica muestra la evolución del porcentaje de cobertura conforme aumentan las 1500 trayectorias en las secciones de Poincaré donde se encontró el porcentaje más alto. La tercera gráfica muestra 1500 rutas planeadas en dichas secciones y finalmente se muestra una gráfica de disperción de las coordenadas visitadas en el área de exploración.

5.1. Diseño de RNGs basados en un solo sistema caótico

Los RNGs diseñados con un solo sistema caótico se mencionan a continuación:

- RNGs basados en el circuito de Chua.
  - RNG basado en el circuito de Chua sin técnicas de post-procesamiento.
  - RNG basado en el circuito de Chua implementando técnicas de post-procesamiento (VN).
  - RNG basado en el sistema de Chua normalizado (L1) para el anáisis de la dependencia del máximo exponente de Lyapunov (Mostrado en el Apéndice A).

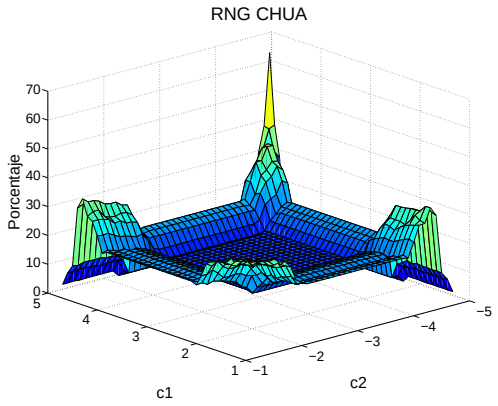
  - RNG basado en el sistema de Chua normalizado (L2) para el anáisis de la dependencia del máximo exponente de Lyapunov (Mostrado en el Apéndice A).
  - RNG basado en el sistema de Chua normalizado (L3) para el anáisis de la dependencia del máximo exponente de Lyapunov (Mostrado en el Apéndice A).
- RNGs basados en el sistema de Lorenz.
  - RNG basado en el sistema de Lorenz con los parámetros utilizados en la literatura.
  - RNG basado en el sistema de Lorenz con el parámetro $\gamma = 28$ (Mostrado en el Apéndice A).
- RNGs basados en una PWL de 2x1.
  - RNG basado en una función saturada de 2x1 (L1) para el anáisis de la dependencia del máximo exponente de Lyapunov.
  - RNG basado en una función saturada de 2x1 (L2) para el anáisis de la dependencia del máximo exponente de Lyapunov.
  - RNG basado en una función saturada de 2x1 (L3) para el anáisis de la dependencia del máximo exponente de Lyapunov.
  - RNG basado en una función saturada de 2x1 con ancho de paso calculados en función de los eigenvalores del sistema (Mostrado en el Apéndice A).
  - RNG basado en una función saturada de 2x1 con un ancho de paso $h = 0.15$ (Mostrado en el Apéndice A).
- RNG basado en una PWL de 4x1.
- RNGs basados en una PWL de 2x2.
  - RNG basado en una función saturada de 2x2 con ancho de paso calculados en función de los eigenvalores del sistema.
  - RNG basado en una función saturada de 2x2 con un ancho de paso $h = 0.15$ (Mostrado en el Apéndice A).
- RNG basado en una PWL de 2x2 Dual.
- RNG basado en una PWL de 4x4 (Mostrado en el Apéndice A).

5.1.1. RNG basado en el circuito de Chua sin la técnica VN

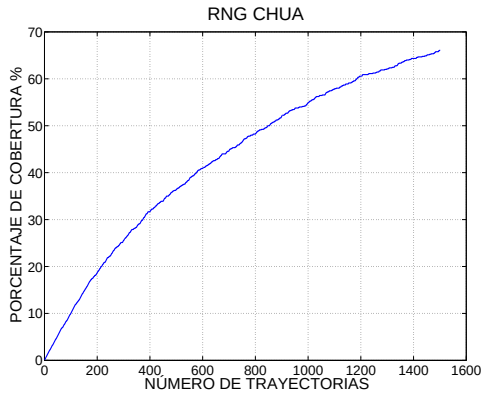
El primer generador de números aleatorios que se propone está basado en el circuito de Chua, al cual se le aplica la técnica de generación de bits descrita en el Capítulo 4 prescindiendo de la técnica de post-procesamiento. La Figura 5.1 (a) muestra la gráfica de superficie de los porcentajes de cobertura después de simular 1500 trayectorias de exploración en cada par de secciones de Poincaré incluidas en el barrido. En el eje x se encuentra la sección de Poincaré

positiva $c1$, en el eje y se encuentra la sección negativa $c2$ y en el eje z los porcentajes de cobertura.

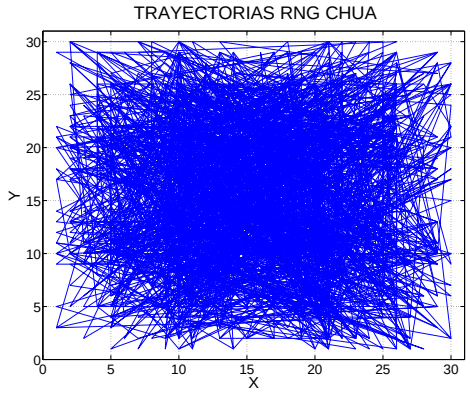
En la gráfica se observa que las secciones de Poincaré en donde se obtuvo el porcentaje más alto fueron ($c1 = 4.7$ y $c2 = -4.7$) con un porcenjate de cobertura de 66.11 %. También se puede observar que en las demás combinaciones el porcentaje de cobertura es muy bajo. En la Figura 5.1 (b) se observa la evolución del porcentaje de cobertura conforme se planean las trayectorias de exploración en las secciones de Poincaré "óptimas", en donde se observa un comportamiento exponencial que se va saturando conforme aumenta el número de trayectorias. En la Figura 5.1 (c) se muestran 1500 trayectorias de exploración que seguiría un robot en un área de 30x30 unidades utilizando el generador de números aleatorios basado en el circuito de Chua sin técnicas de post-procesamiento en las secciones óptimas $c1 = 4.7$, $c2 = -4.7$. Se puede observar que las orillas son poco visitadas y por lo tanto la cobertura no es muy buena, esto lo podemos comprobar con la gráfica de dispersión de la Figura 5.1 (d), en donde se observa el número de veces en que fueron visitadas las coordenadas en en área de exploración.



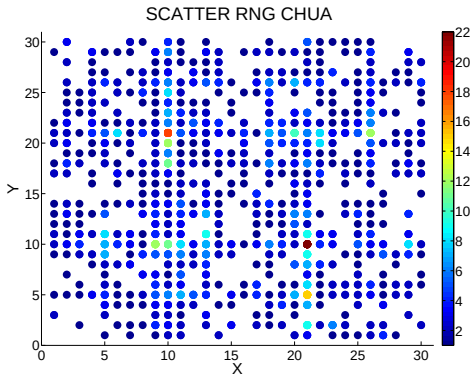
(a) Gráfica de porcentajes.



(b) Porc. de cobertura en $c1 = 4.7$ y $c2 = -4.7$



(c) Trayectorias RNG Chua

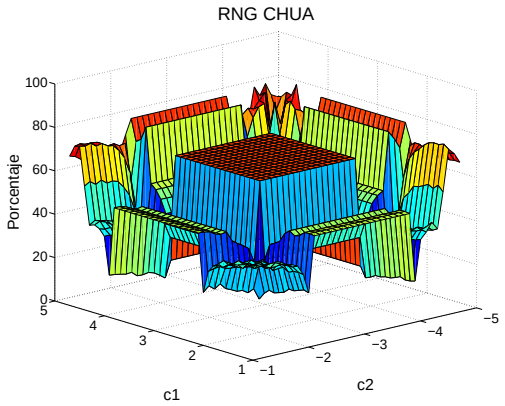


(d) Gráfica de dispersión

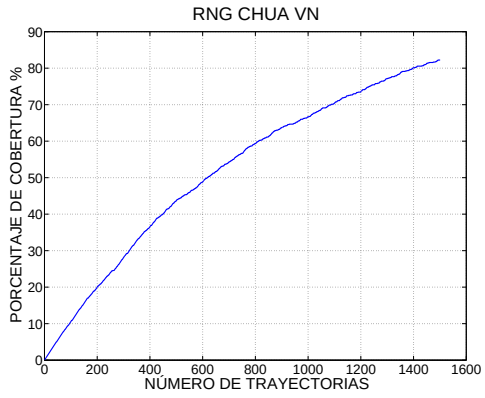
Figura 5.1: Porcentajes y trayectorias planeadas del RNG basado en el circuito de Chua sin la técnica de VN.

5.1.2. RNG basado en el circuito de Chua con la técnica VN

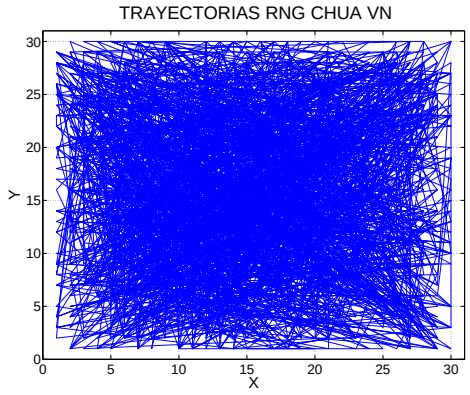
El RNG que se presenta a continuación tambien está basado en el circuito de Chua, la diferencia con el anterior radica en la implementación de la técnica de post-procesamiento de bits de Von Neumann. En la gráfica de la Figura 5.2 (a) se observa que, implementando la técnica VN, los porcentajes de cobertura aumentan considerablemente. Las secciones de Poincaré óptimas en este RNG fueron $c1 = 4.3$, $c2 = -4.7$ con un porcentaje de cobertura del 82.22 %. En la Figura 5.2 (b) se observa la evolución del porcentaje de cobertura conforme transcurren las trayectorias planeadas en las secciones óptimas. En la Figura 5.2 (c) se muestran 1500 trayectorias que seguiría un robot en un área de 30x30 unidades utilizando el generador de números aleatorios basado en el circuito de Chua (VN) en las secciones de corte óptimas, mientras que en (d) se muestra la gráfica de dispersión de las trayectorias planeadas.



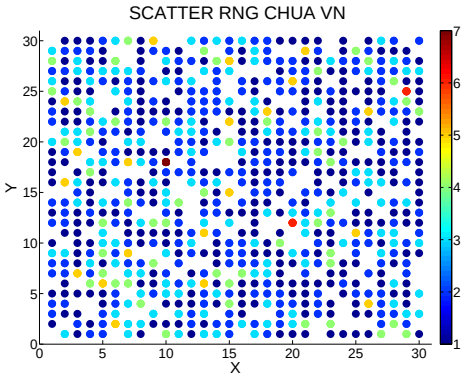
(a) Gráfica de porcentajes.



(b) Porc. de cobertura en $c1 = 4.3$, $c2 = -4.7$.



(c) Trayectorias RNG Chua (VN)



(d) Gráfica de dispersión

Figura 5.2: RNG basado en el circuito de Chua con la técnica de VN.

En la Tabla 5.1 se presentan los resultados de los RNGs basados en el circuito de Chua. Como se puede observar, implementando la técnica de post procesamiento de Von Neumann los porcentajes de cobertura mejoran considerablemente y la repetición en las coordenadas disminuye, aunque esto se refleje tambien en un número mucho mayor de iteraciones debido a los bits descartados por la técnica VN y por tanto un tiempo de cómputo mayor.

Datos	**RNG Chua sin VN**	**RNG Chua sin VN**
Ancho de paso de integración	1×10^7	1×10^7
Porcentaje de cobertura en los puntos de equilibrio	13.1111 %	68.5556 %
Secciones de Poincaré óptimas	$c1 = 4.7$ $c2 = -4.7$	$c1 = 4.3$ $c2 = -4.7$
Porcentaje de cobertura en las secciones óptimas	66.1111 %	82.2222 %
Número de iteraciones para generar 1500 trayectorias	15514580	32443934
Porcentajes mayores a 60 %	1	569
Porcentajes mayores a 70 %	0	167
Porcentajes mayores a 80 %	0	3
Porcentaje promedio	17.0632 %	52.6895 %
Trayectorias necesarias para cubrir el 100 % del área	23238	9301
Iteraciones necesarias para cubrir el 100 % del área	239862635	208016662
Total de bits generados en las secciones óptimas	15550	101392
Bits descartados por la técnica de Von Neumann	–	68692
Bits descartados por estar fuera del límite ($1 \leq x \leq 30$)	550	1350
Máximo exponente de Lyapunov	0.0036	0.0036

Cuadro 5.1: Resultados obtenidos del RGN basado en el circuito de Chua.

Con fines de comprobar la importancia de las técnica de post-procesamiento en la generación de números aleatorios, solo en este sistema se diseñó un generador sin la técnica de VN y otro con dicha técnica, en los siguientes RNGs el análisis se lleva a cabo con la técnica de post-procesamiento de VN.

La interpretación de las gráficas de los RNGs de esta sección es la misma, por lo que en los siguientes RNGs solo se mostrarán las gráficas y la Tabla de resultados.

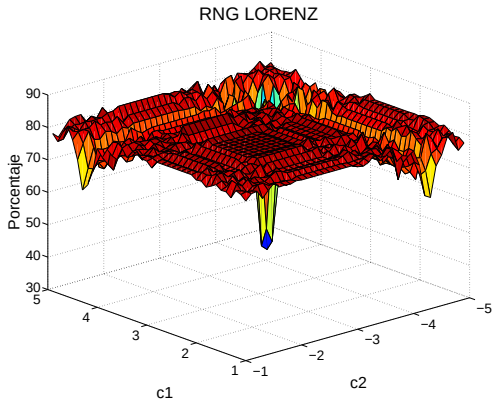
5.1.3. RNG basado en el sistema de Lorenz

En este RNG se utiliza la señal $x(t)$ del sistema de Lorenz como fuente de entropía, en donde $c1 = 3.4$ y $c2 = -2, ..., -2.8$ resultaron ser las secciones de Poincaré con porcentajes de cobertura mas altos con un 84.33 %.

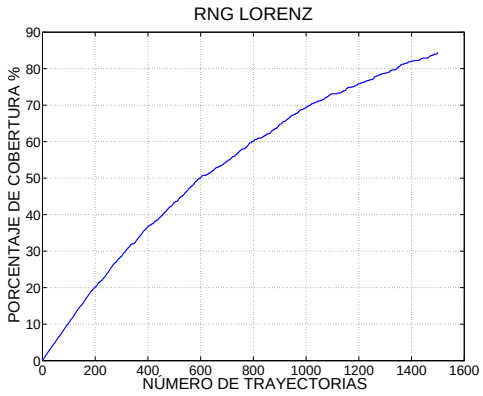
En la Tabla 5.2 se presentan los resultados correspondientes al RNG basado en el sistema de Lorenz.

Datos	**Resultados**
Ancho de paso de integración	0.0037
Porcentaje de cobertura en los puntos de equilibrio	81.8889 %
Secciones de Poincaré óptimas	$c1 = 3.4$ $c2 = -2.5$
Porcentaje de cobertura en las secciones óptimas	84.3333 %
Número de iteraciones para generar 1500 trayectorias en las secciones óptimas	15964878
Porcentajes mayores a 60 %	1586
Porcentajes mayores a 70 %	1548
Porcentajes mayores a 80 %	947
Porcentaje promedio	78.9815 %
Trayectorias necesarias para cubrir el 100 % del área	6718
Iteraciones necesarias para cubrir el 100 % del área	71342450
Total de bits generados en las secciones óptimas	72286
Bits descartados por la técnica de Von Neumann	37566
Bits descartados por estar fuera del límite ($1 \leq x \leq 30$)	2360
Máximo exponente de Lyapunov	0.0029

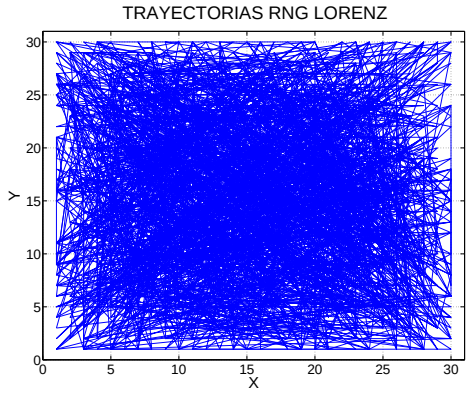
Cuadro 5.2: Resultados obtenidos del RGN basado en el sistema de Lorenz.



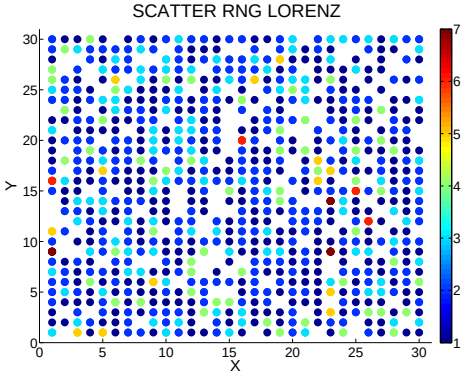
(a) Gráfica de porcentajes



(b) Porc. de cobertura en $c1 = 3.4$ y $c2 = -2.5$



(c) Trayectorias RNG Lorenz



(d) Gráfica de dispersión

Figura 5.3: RNG basado en el sistema de Lorenz.

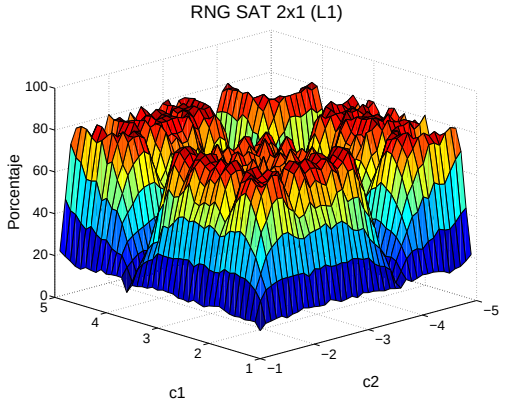
5.1.4. RNG función saturada 2x1 (L1)

En este RNG se utiliza la señal $x(t)$ del sistema basado en una función saturada de 2x1 como fuente de entropía con los parámetros $a = b = c = d = 0.7$.

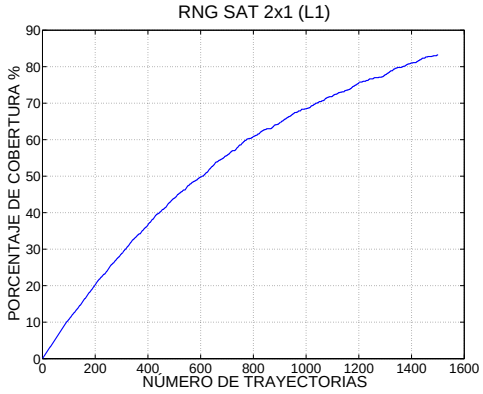
Datos	**Resultados**
Ancho de paso	0.0152
Porcentaje de cobertura en los puntos de equilibrio	66.1111 %
Secciones óptimas	$c1 = 3.2$ $c2 = -4.4$
Porcentaje de cobertura en las secciones óptimas	83.2222 %
Número de iteraciones para generar 1500 trayectorias en las secciones óptimas	55177100
Porcentajes mayores a 60 %	883
Porcentajes mayores a 70 %	603
Porcentajes mayores a 80 %	143
Porcentaje promedio	57.4542 %
Trayectorias necesarias para cubrir el 100 % del área	6898
Iteraciones necesarias para cubrir el 100 % del área	250398972
Total de bits generados en las secciones óptimas	60548
Bits descartados por la técnica de Von Neumann	25208
Bits descartados por estar fuera del límite ($1 \leq x \leq 30$)	2670
Máximo exponente de Lyapunov	0.0011

Cuadro 5.3: Resultados obtenidos del RGN basado en el sistema PWL de 2x1 (L1).

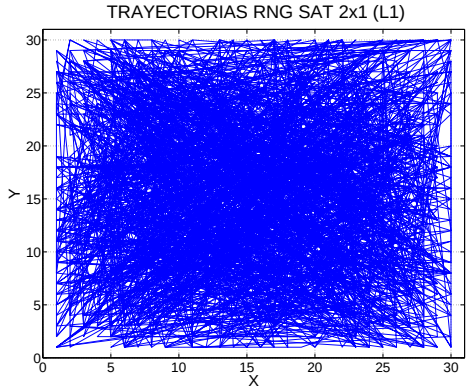
En las secciones $c1 = 3.2$ y $c2 = -4.4$ se obtiene el porcentaje de cobertura más alto con valor de 83.2222 %.



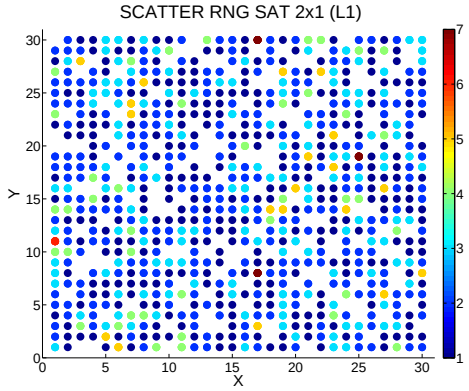
(a) Gráfica de porcentajes.



(b) Porc. de cobertura en $c1 = 3.2$ y $c2 = -4.4$.



(c) Trayectorias de exploración



(d) Gráfica de dispersión

Figura 5.4: RNG del sistema basado en una función saturada de 2x1 (L1).

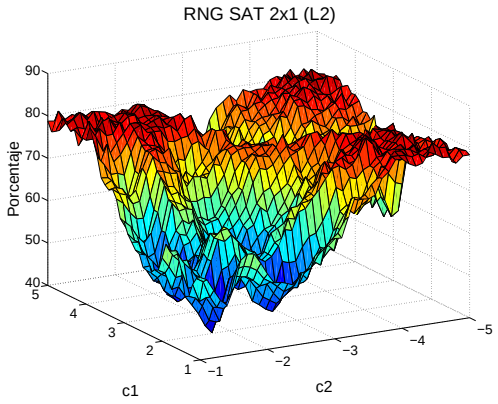
5.1.5. RNG función saturada 2x1 (L2)

En este RNG se utiliza la señal $x(t)$ del sistema basado en una función saturada como entrada igual que el anterior, la diferencia radica en el cambio en los parámetros del sistema para optimizar el máximo exponente de Lyapunov.

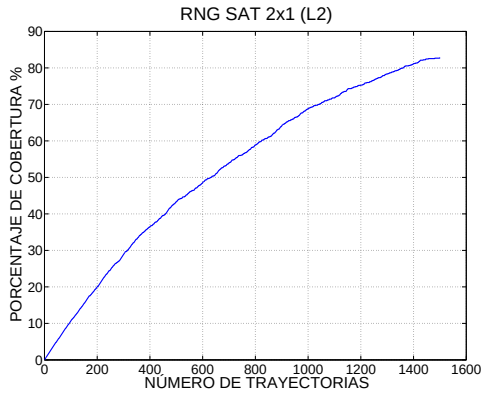
Para incrementar el valor del máximo exponente de Lyapunov se ajustan los parámetros del sistema de la siguiente manera: $a = 1, b = 1, c = 0.499, d = 1$. Los resultados revelan que el porcentaje de cobertura mas alto que se obtuvo después de realizar el barrido en la señal $x(t)$ se encuentra en las secciones $c1 = 1.6$ y $c2 = -3.8$ con 82.7778 %. En la Tabla 5.4 se presentan los resultados del RNG basado en una funciós saturada, el cuál cuenta con la primera optimización del máximo exponente de Lyapunov (L2).

Datos	**Resultados**
Ancho de paso	0.0152
Porcentaje de cobertura en los puntos de equilibrio	53.1111 %
Secciones óptimas	$c1 = 1.6$ $c2 = -3.8$
Porcentaje de cobertura en las secciones óptimas	82.7778 %
Número de iteraciones para generar 1500 trayectorias	24242747
Porcentajes mayores a 60 %	1314
Porcentajes mayores a 70 %	875
Porcentajes mayores a 80 %	146
Porcentaje promedio	69.1299 %
Trayectorias necesarias para cubrir el 100 % del área	6148
Iteraciones necesarias para cubrir el 100 % del área	100488992
Total de bits generados en las secciones óptimas	50270
Bits descartados por la técnica de Von Neumann	17430
Bits descartados por estar fuera del límite ($1 \leq x \leq 30$)	1420
Máximo exponente de Lyapunov	0.0018

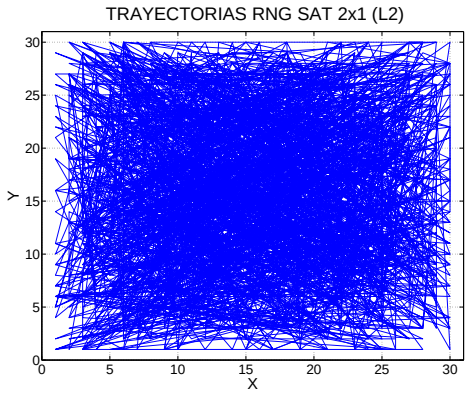
Cuadro 5.4: Resultados del RGN basado en el sistema caótico PWL de 2x1 (L2).



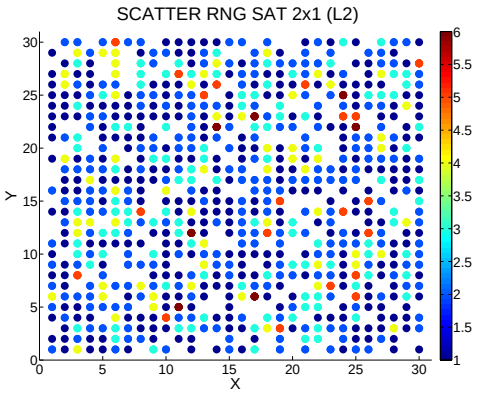
(a) Gráfica de porcentajes.



(b) Porc. de cobertura en $c1 = 1.6$ y $c2 = -3.8$.



(c) Trayectorias de exploración.



(d) Gráfica de dispersión.

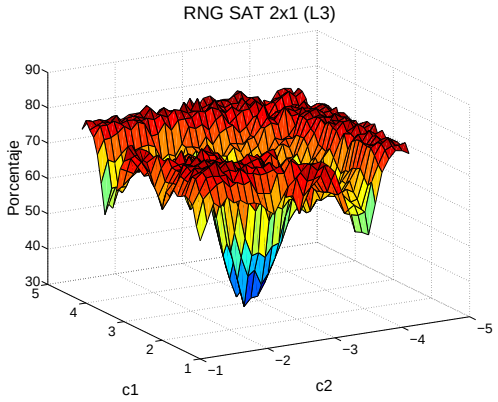
Figura 5.5: RNG basado en una función saturada de 2x1 (L2).

5.1.6. RNG función saturada 2x1 (L3)

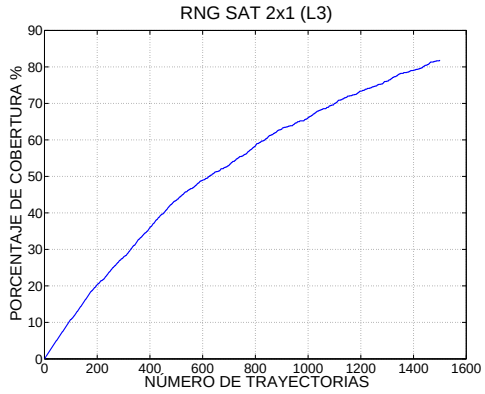
En este RNG se utiliza la señal $x(t)$ del sistema basado en una función saturada como fuente de entropía, en este caso los parámetros del sistema fueron modificados de la siguiente manera: $a = 1, b = 0.788, c = 0.643, d = 0.666$.

Datos	**Resultados**
Ancho de paso	0.0152
Porcentaje de cobertura en los puntos de equilibrio	27.8889 %
Secciones óptimas	$c1 = 3.9$ $c2 = -2.6$
Porcentaje de cobertura en las secciones óptimas	83.6667 %
Número de iteraciones para generar 1500 trayectorias en las secciones óptimas	106677015
Porcentajes mayores a 60 %	935
Porcentajes mayores a 70 %	760
Porcentajes mayores a 80 %	246
Porcentaje promedio	73.2444 %
Trayectorias necesarias para cubrir el 100 % del área	7391
Iteraciones necesarias para cubrir el 100 % del área	439183062
Total de bits generados en las secciones óptimas	126086
Bits descartados por la técnica de Von Neumann	90886
Bits descartados por estar fuera del límite ($1 \leq x \leq 30$)	2600
Máximo exponente de Lyapunov	0.0031

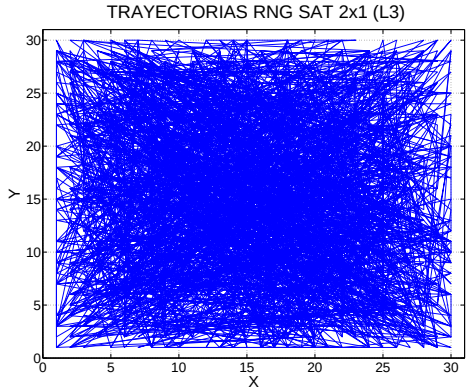
Cuadro 5.5: Resultados del RGN del sistema caótico basado en una función saturada 2x1 (L3).



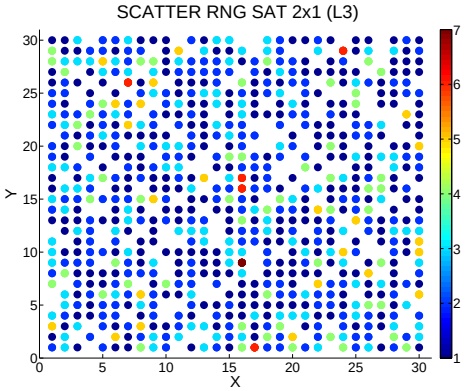
(a) Gráfica de porcentajes.



(b) Porc. de cobertura en $c1 = 3.9$ y $c2 = -2.6$.



(c) Trayectorias de exploración



(d) Gráfica de dispersión

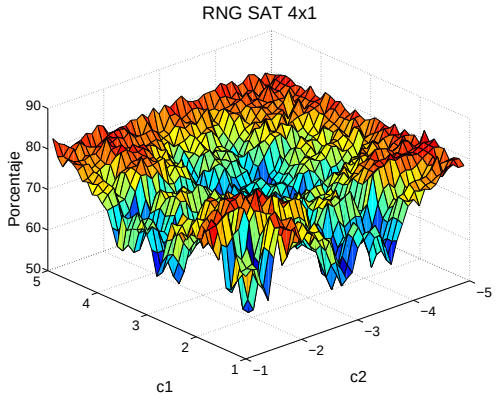
Figura 5.6: RNG del sistema basado en una función saturada de 2x1 (L3).

5.1.7. RNG del sistema basado en una función saturada de 4x1

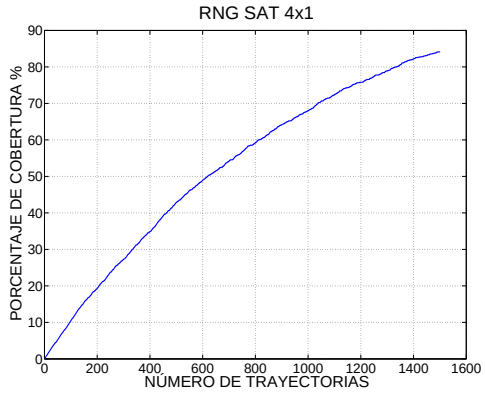
Este RNG utiliza la señal $x(t)$ del sistema basado en una función saturada de cuatro enrollamientos en una dimensión (4x1) en la generación de números aleatorios.

Datos	**Resultados**
Ancho de paso de integración	0.1102
Porcentaje de cobertura en los puntos de equilibrio	71.2222 %, 75.6667 %, 68.1111 %
Secciones óptimas	$c1 = 1.7$ $c2 = -1.8$
Porcentaje de cobertura en las secciones óptimas	84.1111 %
Número de iteraciones para generar 1500 trayectorias en las secciones óptimas	15899497
Porcentajes mayores a 60 %	1584
Porcentajes mayores a 70 %	1158
Porcentajes mayores a 80 %	142
Porcentaje promedio	73.38 %
Trayectorias necesarias para cubrir el 100 % del área	6863
Iteraciones necesarias para cubrir el 100 % del área	73310237
Total de bits generados en las secciones óptimas	90114
Bits descartados por la técnica de Von Neumann	55034
Bits descartados por estar fuera del límite $(1 \leq x \leq 30)$	2540
Máximo exponente de Lyapunov	0.0157

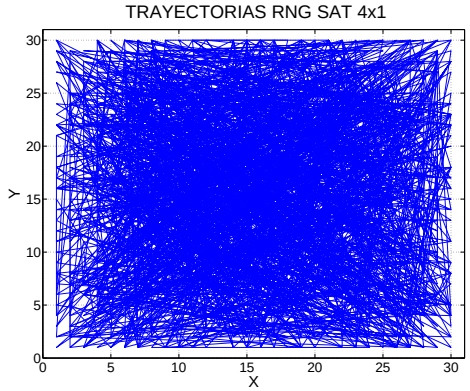
Cuadro 5.6: Resultados obtenidos del RGN basado en el sistema basado en una serie de funciones saturadadas de 4x1.



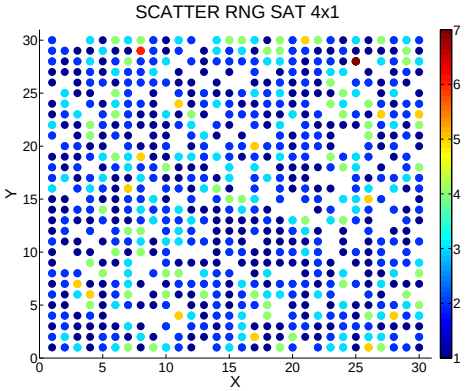
(a) Gráfica de porcentajes.



(b) Porc. de cobertura en $c1 = 1.7$ y $c2 = -1.8$.



(c) Trayectorias RNG Saturada 4x1.



(d) Gráfica de dispersión.

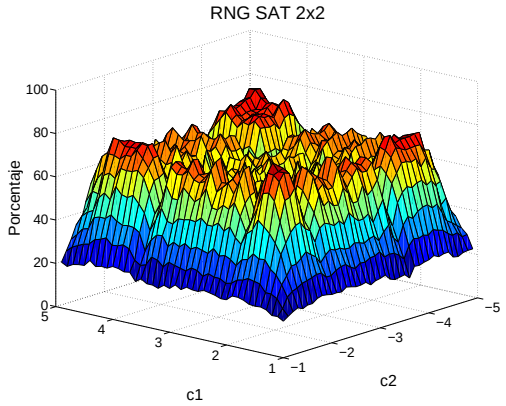
Figura 5.7: RNG del sistema basado en una función saturada 4x1.

5.1.8. RNG del sistema basado en una función saturada de 2x2

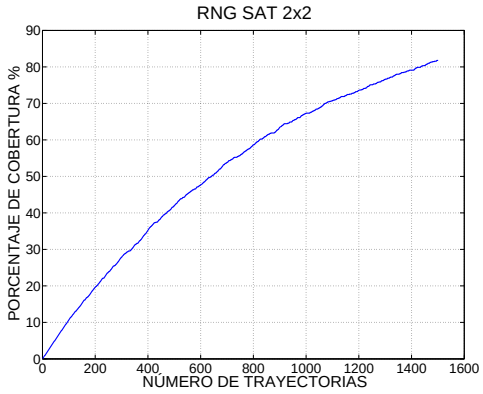
En este RNG la fuente de entropía está dada por la señal $x(t)$ del sistema basado en una función saturada con dos enrollamientos en 2D. Las secciones con el porcentaje de cobertura mas alto se ubican en $c1 = 1.7 y c2 = -1.6$ con 81.7777 %.

Datos	**Resultados**
Ancho de paso de integración	0.0152
Porcentaje de cobertura en los puntos de equilibrio	58.3333 %
Secciones óptimas	$c1 = 1.7$ $c2 = -1.6$
Porcentaje de cobertura en las secciones óptimas	81.7777 %
Número de iteraciones para generar 1500 trayectorias en las secciones óptimas	51445000
Porcentajes mayores a 60 %	737
Porcentajes mayores a 70 %	220
Porcentajes mayores a 80 %	4
Porcentaje promedio	56.0615 %
Trayectorias necesarias para cubrir el 100 % del área	6893
Iteraciones necesarias para cubrir el 100 % del área	238054235
Total de bits generados en las secciones óptimas	81762
Bits descartados por la técnica de Von Neumann	49002
Bits descartados por estar fuera del límite $(1 \leq x \leq 30)$	1380
Máximo exponente de Lyapunov	0.000829

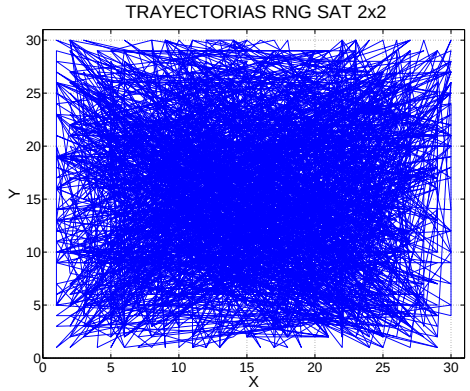
Cuadro 5.7: Resultados obtenidos del RGN basado en el sistema basado en una función saturada 2x2.



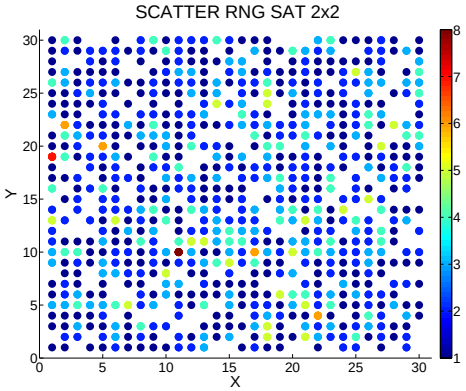
(a) Gráfica de porcentajes.



(b) Porc. de cobertura en $c1 = 1.7$ y $c2 = -1.6$.



(c) Trayectorias RNG Saturada 2x2



(d) Gráfica de dispersión

Figura 5.8: RNG basado en una función saturada 2x2.

5.1.9. RNG Dual

En este RNG se utiliza la señal $x(t)$ y $y(t)$ del sistema basado en una función saturada de 2x2 para la obtención simultánea de números aleatorios, esto es, se tendrán dos generadores de números aleatorios a partir de un solo sistema. Para ello, se toman las secciones de Poincaré de la señal $x(t)$ donde se encontró el porcentaje de cobertura mayor y se aplican también para la señal $y(t)$.

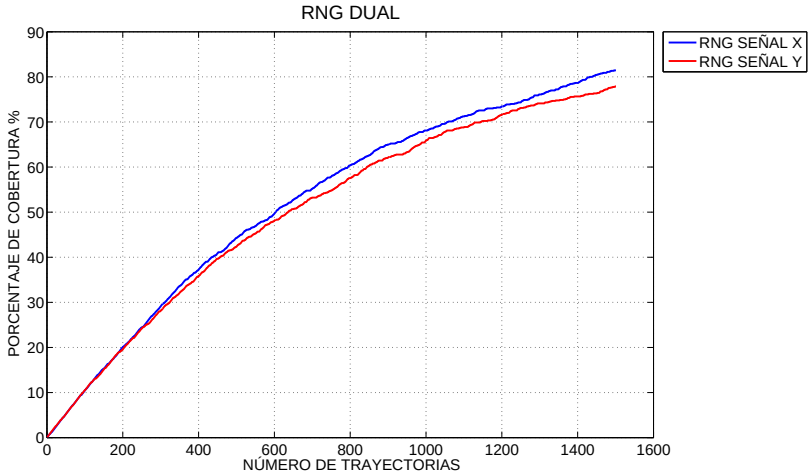


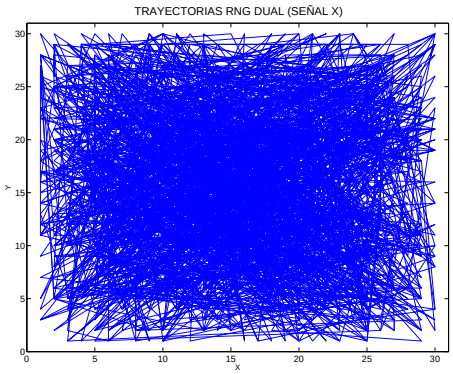
Figura 5.9: Gráfica de porcentajes del RNG Dual.

En la Figura 5.9 se observa el aumento del porcentaje de cobertura a lo largo de las 1500 coordenadas generadas a partir de la señal $x(t)$ (línea azul) y por la señal $y(t)$ (línea roja), en donde los porcentajes de cobertura fueron 81.444 % y 77.8889 % respectivamente.

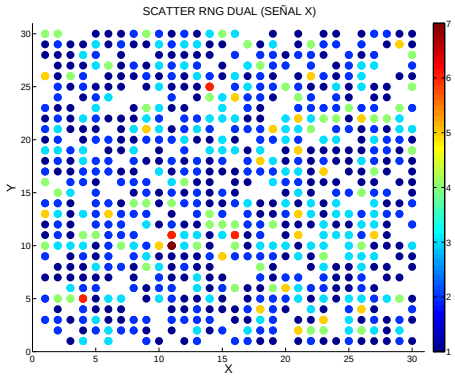
La Figura 5.10 Muestra la trayectoria que seguiría un robot móvil despues de planear 1500 coordenadas en un área de exploración de 30x30 unidades utilizando el generador de números aleatorios basado en la señal $x(t)$ y $y(t)$ de una función saturada de 2x2 en las secciones de Poincaré óptimas, así como las gráficas de dispersión que muestran la distribución de las coordenadas durante la exploración. Los resultados del RNG dual son mostrados en la Tabla 5.8.

Datos	**Señal** $x(t)$	**Señal** $y(t)$
Ancho de paso de integración	0.0152	
Porcentaje de cobertura en los puntos de equilibrio	58.3333 %	63.4444
Secciones óptimas $x(t)$	($c1 = 1.7$	$c1 = -1.6$)
Porcentaje de cobertura en las secciones óptimas	81.7777 %	78.8889 %
Iteraciones para generar 1500 trayectorias	117023652	
Trayectorias para cubrir el 100 % del área	6893	11620
Iteraciones necesarias para cubrir el 100 % del área	913773658	
Total de bits generados en las secciones óptimas	81762	168222
Bits descartados por la técnica de Von Neumann	49002	136122
Bits descartados por estar fuera del límite	1380	1050

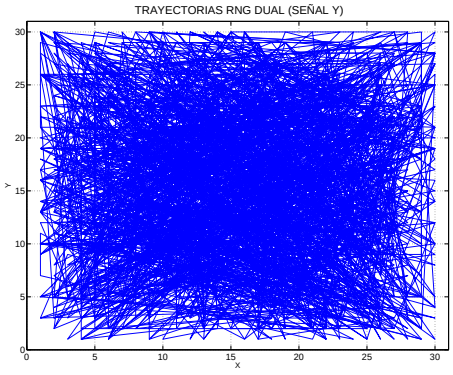
Cuadro 5.8: Resultados obtenidos del RGN dual.



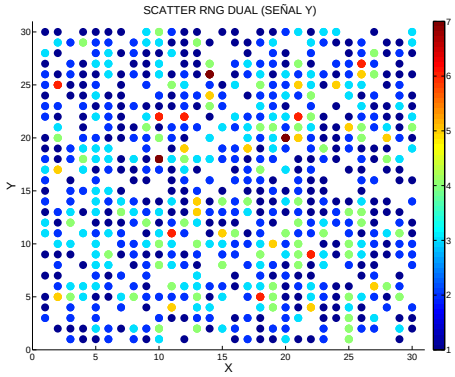
(a) Trayectorias señal x



(b) Gráfica de dispersión (señal $x(t)$)



(c) Trayectorias señal y



(d) Gráfica de dispersión (señal $y(t)$)

Figura 5.10: Trayectorias planeadas por el RNG dual.

5.2. Diseño de RNGs combinando 2 sistemas caóticos a diferente frecuencia

En esta sección se muestran los resultados de los RNGs diseñados a partir de 2 señales caóticas de diferentes sistemas a diferente frecuencia. Se diseñaron un total seis RNGs de este tipo en base a las combinaciones que se muestran en la Tabla 5.9.

RNG	**Señal lenta**	**Señal rápida**
Chua-Lorenz	Chua	Lorenz
Chua-Saturada	Chua	Saturada
Lorenz-Chua	Lorenz	Chua
Lorenz-Saturada	Lorenz	Saturada
Saturada-Chua	Saturada	Chua
Saturada-Lorenz	Saturada	Lorenz

Cuadro 5.9: RNGs diseñados con dos sistemas caóticos a diferente frecuencia

Como se mencionó anteriormente se realizaron una serie de pruebas para determinar el factor de escalamiento (FS) entre la señal lenta $x_1(t)$ y la señal rápida $x_2(t)$ en cada uno de los 6 RNG simulando 1500 puntos de exploración. La prueba se realizó considerando un cruce por cero unidireccional y bidireccional. El FS y el flanco óptimo que se obtuvieron en cada RNG son remarcados en su respectiva tabla. Las gráficas de porcentaje de cobertura en las FS óptimas, las gráficas de las rutas planeadas y las gráficas de las coordenadas visitadas en el área delimitada correspondientes a cada RNG son mostradas en el Apéndice B.

5.2.1. RNG Chua-Lorenz

Señal lenta	Señal rápida	Flanco	FS	Porcentaje	Iteraciones
Chua	Lorenz	**Subida**	**1 − 1**	**80.6667** %	6368851
			1 − 2	80.6667 %	6368851
			1 − 10	81.6667 %	6461100
			1 − 100	80.2222 %	6458137
			1 − 1000	83.2222 %	6486374
			1 − 10000	82.3333 %	6461100
			1 − 100000	82.0000 %	6426698
		Subida y bajada	1 − 1	67.7778 %	3838963
			1 − 2	67.7778 %	3838963
			1 − 10	71.0000 %	3884826
			1 − 100	71.5556 %	3888923
			1 − 1000	69.1111 %	3882787
			1 − 10000	68.6667 %	3799614
			1 − 100000	68.2222 %	3857184

Cuadro 5.10: Resultados obtenidos del RGN Chua-Lorenz.

5.2.2. RNG Chua-Saturada

Señal lenta	Señal rápida	Flanco	FS	Porcentaje	Iteraciones
Chua	Saturada	**Subida**	**1 − 1**	**46.0000** %	7434254
			1 − 2	46.0000 %	7434254
			1 − 10	44.5556 %	7592461
			1 − 100	46.3333 %	7485751
			1 − 1000	45.1111 %	7313483
			1 − 10000	44.1111 %	7500931
			1 − 100000	44.7778 %	7368582
		Subida y bajada	1 − 1	21.1111 %	7931875
			1 − 2	21.1111 %	7931875
			1 − 10	20.4444 %	7852686
			1 − 100	21.6667 %	8388206
			1 − 1000	22.0000 %	7828157
			1 − 10000	21.4444 %	7950015
			1 − 100000	20.6667 %	8161277

Cuadro 5.11: Resultados obtenidos del RGN Chua-Saturada.

5.2.3. RNG Lorenz-Chua

Señal lenta	Señal rápida	Flanco	FS	Porcentaje	Iteraciones
Lorenz	Chua	**Subida**	$1-1$	79.0000 %	16124390
			$1-2$	79.0000 %	16124390
			$1-10$	81.5556 %	16024277
			$1-100$	81.1111 %	16244812
			$1-1000$	**82.0000** %	15879188
			$1-10000$	80.2222 %	16069165
			$1-100000$	82.5556 %	15696164
		Subida y bajada	$1-1$	76.7778 %	7426873
			$1-2$	76.7778 %	7426873
			$1-10$	78.2222 %	7424512
			$1-100$	77.5556 %	7417969
			$1-1000$	76.6667 %	7378679
			$1-10000$	75.5556 %	7312454
			$1-100000$	75.7778 %	7385095

Cuadro 5.12: Resultados obtenidos del RGN Lorenz-Chua.

5.2.4. RNG Lorenz-Saturada

Señal lenta	Señal rápida	Flanco	FS	Porcentaje	Iteraciones
Lorenz	Saturada	**Subida**	**$1-1$**	**80.1111** %	14619488
			$1-2$	80.1111 %	14619488
			$1-10$	79.6667 %	14915974
			$1-100$	79.2222 %	14829768
			$1-1000$	79.2222 %	14838477
			$1-10000$	79.2222 %	14724559
			$1-100000$	80.2222 %	14805759
		Subida y bajada	$1-1$	55.0000 %	8880510
			$1-2$	55.0000 %	8880510
			$1-10$	54.5556 %	9200998
			$1-100$	55.3333 %	9241023
			$1-1000$	56.4444 %	8905892
			$1-10000$	54.8889 %	8920617
			$1-100000$	54.4444 %	9039340

Cuadro 5.13: Resultados obtenidos del RGN Lorenz-Saturada.

5.2.5. RNG Saturada-Chua

Señal lenta	Señal rápida	Flanco	FS	Porcentaje	Iteraciones
Saturada	Chua	Subida	$1-1$	81.0000 %	34093079
			$1-2$	81.0000 %	34093079
			$1-10$	79.2222 %	34284573
			$1-100$	82.5556 %	33701653
			$1-1000$	79.8889 %	34414889
			$1-10000$	81.4444 %	34851891
			$1-100000$	81.6667 %	34153858
		Subida y bajada	**$1-1$**	**81.5556** %	17983309
			$1-2$	81.5556 %	17983309
			$1-10$	80.5556 %	18177212
			$1-100$	76.3333 %	18003419
			$1-1000$	81.7778 %	17741458
			$1-10000$	80.7778 %	17833318
			$1-100000$	80.1111 %	17833318

Cuadro 5.14: Resultados obtenidos del RGN Saturada-Chua.

5.2.6. RNG Saturada-Lorenz

Señal lenta	Señal rápida	Flanco	FS	Porcentaje	Iteraciones
Saturada	Lorenz	Subida	1	81.1111 %	34193865
			2	81.1111 %	34193865
			10	80.0000 %	34546948
			100	81.3333 %	34193865
			1000	81.5556 %	33357953
			10000	80.2222 %	34037758
			100000	81.8889 %	34325187
		Subida y bajada	**1 − 1**	**79.3333** %	17594335
			2	79.3333 %	17594335
			10	81.0000 %	17621215
			100	80.5556 %	18076026
			1000	80.5556 %	17511491
			10000	80.1111 %	17761627
			100000	80.2222 %	17231613

Cuadro 5.15: Resultados obtenidos del RGN Saturada-Lorenz.

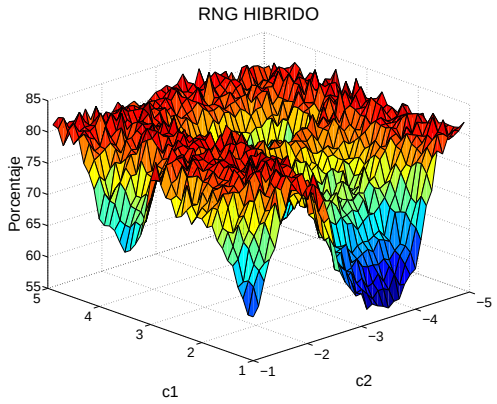
5.3. Diseño del RNG híbrido

En este sistema híbrido, el sistema de Chua será el encargado de alternar entre el sistema de Lorenz y el sistema basado en una función saturada de dos enrollamientos para la generación de números aleatorios dercrito en la sección 3.1.3. Una vez que se tiene la señal combinada es necesario realizar el proceso de barrido para determinar las secciones de Poincaré óptimas en función al porcentaje de cobertura.

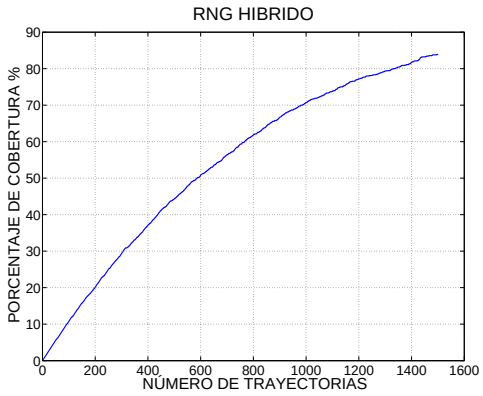
En la gráfica de la Figura 5.11 (a) se observan los porcentajes correspondientes a todas las secciones incluidas, en donde $c1 = 3$ y $c2 = -4.8$ resultan ser las secciones óptimas ya que se obtiene el porcentaje de cobertura más alto con valor de 83.8889 %. En la Figura 5.11 (b) se observa la evolución del porcentaje de cobertura a lo largo de las 1500 trayectorias en las secciones de corte óptimas. En la Figura 5.11 (c) se muestran 1500 trayectorias planeadas por el generador de números aleatorios basado en sistema híbrido en las secciones de Poincaré óptimas, mientras que en la Figura 5.11 (d) se muestra el número de veces que fueron visitadas las coordenadas en el área de exploración. En la Tabla 5.16 se presentan los datos más importantes.

Datos	Resultados
Secciones óptimas	$c1 = 3$ $c2 = -4.8$
Porcentaje de cobertura en las secciones óptimas	83.8889 %
Número de iteraciones para generar 1500 trayectorias en las secciones óptimas	143301074
Porcentajes mayores a 60 %	1583
Porcentajes mayores a 70 %	1355
Porcentajes mayores a 80 %	516
Porcentaje promedio	76.3085 %
Trayectorias necesarias para cubrir el 100 % del área	9905
Iteraciones necesarias para cubrir el 100 % del área	939524096
Total de bits generados en las secciones óptimas	320676
Bits descartados por la técnica de Von Neumann	285916
Bits descartados por estar fuera del límite ($1 \leq x \leq 30$)	2380
Máximo exponente de Lyapunov	0.0007524

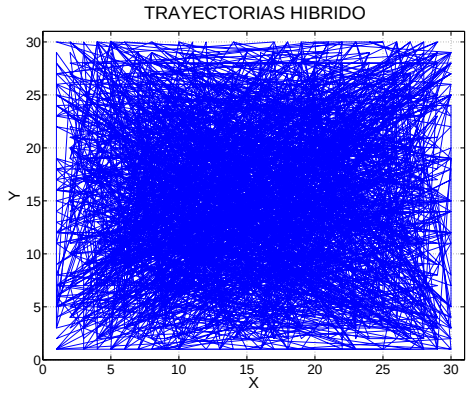
Cuadro 5.16: Resultados obtenidos del RGN basado en un sistema híbrido.



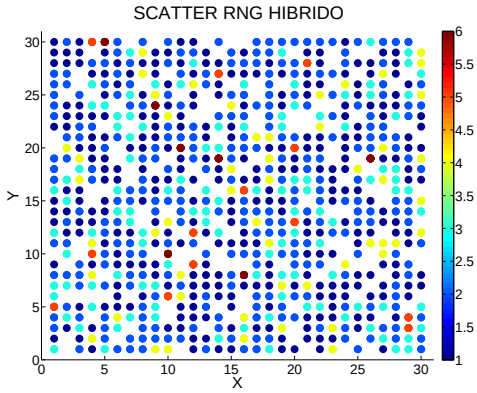
(a) Gráfica de porcentajes.



(b) Porc. de cobertura en $c1 = 3$ y $c2 = -4.8$.



(c) Trayectorias RNG híbrido.



(d) Gráfica de dispersión.

Figura 5.11: Porcentajes RNG del sistema híbrido.

Capítulo 6

Análisis de resultados

En este capítulo se discuten los resultados obtenidos de todos los RNGs diseñados para la generación de rutas de exploración. Para ello, el análisis de se divide en tres secciones: En primer lugar se discuten los RNGs diseñados con un solo sistema, después se analizan los resultados de los RNGs combinados y finalmente los resultados correspondientes al estudio de la dependencia del máximo exponente de Lyapunov en la generación de números aleatorios. Las pruebas estadísticas de aleatoriedad se llevaron a cabo mediante el paquete de pruebas estadísticas NIST SP-800.22, el cuál es descrito en el Apéndice D, así como las tablas con los *p-values* de cada prueba.

6.1. RNGs con un solo sistema caótico

En primer lugar se ha demostrado la importancia de las técnicas de post-procesamiento en el diseño de generadores de números aleatorios. En partícular se demuestra que implementando la técnica de Von Neumman la distribución de los bits se mejora de forma importante, lo que reduce la redundancia en la generación de números y por consiguiente el porcentaje de cobertura se incrementa considerablemente.

Se han presentado diferentes versiones los RNGs basados en el sistema de Lorenz y en una función saturada de 2 y 4 enrollamientos, con el objetivo de deducir si existe alguna correlación entre la variación de los parámetros del sistema o el ancho de paso de integración y el porcentaje de cobertura. Particularmente se desarrollaron dos versiones del RNG basado en el sistema de Lorenz: La primera versión (v1) se integra con los valores $a = 10, b = 8/3, c = 24$, la segunda versión (v2) se integra con los valores $a = 10, b = 8/3, c = 28$, ambas con el mismo ancho de paso de integración. Como resultado no se observaron cambios significativos en el porcentaje de cobertura ni en la pruebas estadísticas, por lo que se concluye que este sistema persiste su uniformidad ante la variación el los parámetros del sistema con esta técnica en particular para generar bits aleatorios.

Por otra parte, se propusieron dos versiones del RNG basado en una función saturada de dos enrollamientos (2x1): la primera versión (v1) consta de una pendiente de 9 en la PWL y un ancho de paso de integración calculado en función de los eigenvalores del sistema, la segunda versión (v2) consta de una pendiente de 100 en la PWL y un ancho de paso de integración mas grande ($h = 0.15$). En este sistema se encontró un cambio significativo en la uniformidad de los

porcentajes así como en los porcentajes promedio al variar la pendiente de la función PWL y el ancho de paso de integración, por lo que se concluye que la eficiencia de los RNGs basados en este sistema en particular depende fuertemente de la construcción de la función PWL ya que afecta de forma importante la dinámica de los sistemas.

RNG / DATOS	Chua	Chua VN	Lorenz v1	Lorenz v2	Saturada 2x1 v1	Saturada 2x1 v2	Saturada 2x2 v1	Saturada 2x2 v2	Saturada 4x1	Saturada 4x4	Sistema Híbrido
Porcentaje en los Puntos de Equilibrio	13.1111%	68.5556%	81.8889%	80.7778%	54.0000%	63.7778%	58.3333%	57.8889%	75.6667%	72.3333%	--------
Secciones óptimas	c1=4.7 c2=-4.7	c1=4.3 c2=-4.7	c1=3.4 c2=-2.5	c1=4.9 c2=-2.1	c1=4.8 c2=-2.5	c1=4.3 c2=-4.9	c1=1.7 c2=-1.6	c1=2.9 c2=-2.7	c1=1.7 c2=-1.8	c1=2.1 c2=-2	c1=3 c2=-4.8
Porcentaje en las secciones óptimas	66.1111%	82.2222%	84.3333%	84.8888%	83.5555%	83.7777%	81.7777%	82.8888%	84.1111%	82.4444%	82.4444%
No. de iteraciones para generar 1500 trayectorias	15514580	32443934	15964878	137729533	123962019	45986500	51445000	2601018	15899497	102238110	143301074
Porcentajes por encima del 60%	1	539	1586	1579	881	1386	737	736	1584	1099	1583
Porcentajes por encima del 70%	0	167	1548	1507	568	1084	220	313	1158	227	1355
Porcentajes por encima del 80%	0	3	947	823	181	325	4	25	142	15	516
Porcentaje promedio	17.0632%	52.6895%	78.9815%	78.5033%	59.4619%	71.972%	56.0615%	58.7167%	73.38%	62.976%	76.3085%
Bits para generar 1500 trayectorias en las secciones óptimas	15550	101392	72286	371088	28099528	104872	81762	48776	90114	82578	320676
Bits descartados por la técnica VN	-----	68692	37566	337408	28065648	71412	49002	14436	55034	49538	285916
Bits descartados por estar fuera del llimite	550	1350	2360	1840	1940	1730	1380	2170	2540	1520	2380
Pruebas NIST exitosas con 10,000 bits	4/8	5/8	5/8	5/8	5/8	5/8	5/8	5/8	7/8	5/8	5/8
Pruebas NIST exitosas con 100,000 bits	4/9	6/9	6/9	6/9	6/9	6/9	5/9	6/9	6/9	6/9	6/9
Pruebas NIST exitosas con 1,000,000 de bits	5/15	8/15	6/15	8/15	8/15	8/15	5/15	8/15	6/15	6/15	----

Figura 6.1: Tabla comparativa de los RNGs con un solo sistema.

También se diseñaron dos versiones del sistema basado en series de funciones saturadas de 2 enrollamientos en 2 dimensiones (2x2), la diferencia entre estas dos versiones es el cambio en el ancho de paso de integración h: en la primera versión (v1) h está calculada en función de los eigenvalores del sistema, mientras que la segunda versión (v2) el ancho de paso de integración es definido en $h = 0.15$. En este caso la uniformidad y el promedio de los porcentajes de cobertura no se ve muy afectado por el cambio en el ancho de paso de integración, sin embargo en las gráficas de cobertura se puede observar que la ubicación de secciones de Poincaré que se podrían considerar como óptimas cambian radicalmente de ubicación, por lo que se debe de tener cuidado al momento de calcular el ancho de paso de integración.

Como se mencionó anteriormente se determinaron como punto de partida las secciones de Poincaré "óptimas" en función del porcentaje de cobertura mayor de cada RNG, sin embargo se realizaron una serie de pruebas con las cuales se concluye que cualquier sección de Poincaré que se encuentre cerca o por encima del 80 % se puede considerar como una combinación óptima

para la obtención de bits aleatorios y más aún si las secciones de corte se encuentran situadas en la parte media de la señal, ya que esto asegura que los bits se obtengan más rápidamente y exista una menor cantidad de bits desechados por la técnica de VN como se muestra en la Tabla comparativa.

Debido que la mayoría de los RNGs nos proporcionan porcentajes cercanos o mayores a 80 %, una forma de concluir que un RNG es más eficiente que el otro sería por medio del análisis de los porcentaje promedio y por medio de las gráficas de superficie, las cuales nos dan una idea de la uniformidad que existe en los porcentajes de cobertura durante el barrido de las secciones de Poincaré.

Con este criterio para la evaluación en la eficiencia de este primer bloque de RNGs, el sistema de Lorenz presentó un porcentaje de cobertura promedio mayor, más uniformidad que los demás y un número mayor de secciones con porcentajes que rondan o superan el 80 % de cobertura. Sin embargo existe algo que irremediablemente se tiene que considerar: la implementación en hardware. Implementar mediante circuitos electrónicos el sistema de Lorenz es mucho más complicado que el sistema de Chua o el sistema basado en una función PWL debido a que la no linealidad de este sistema está determinada por la multiplicación de variables de estado, y trabajar con multiplicadores analógicos es más complicado y costoso. Es por eso que el RNG basado en un sistema caótico PWL resulta ser el más conveniente.

Por otra parte se diseñó también un RNG dual aprovechando la dinámica compleja de las señales x(t) y y(t) que nos proporciona un sistema basado en una serie de funciones saturadas de 2x2. Para ello se obtuvieron las secciones de Poincaré determinadas con anterioridad de la señal x(t) y se aplicaron para y(t) para obtener dos fuentes para generar bits en un solo sistema. Como resultado se obtiene un RNG bastante interesante ya que se aprovecha de una mejor forma la dinámica que ofrecen las dos señales del mismo sistema.

6.2. RNGs combinado dos sistemas caóticos a diferente frecuencia

En la segunda sección se diseñaron seis RNGs con 2 señales caóticas a diferente frecuencia, para ello se hizo un estudio para determinar el flanco de la señal lenta y el factor de escalamiento óptimo de la señal rápida. Los RNGs diseñados con esta técnica fueron:

- RNG Chua-Lorenz.
- RNG Chua-Saturada.
- RNG Lorenz-Chua.
- RNG Lorenz-Saturada.
- RNG Saturada-Chua.
- RNG Saturada-Lorenz.

Cabe destacar que en los RNGs antes mencionados NO SE IMPLEMENTÓ LA TÉCNICA DE VON NEUMANN ya que la mayoría presentaron porcentajes de cobertura altos sin necesidad de implementar un post-procesamiento.

Los resultados mostrados en la Tabla comparativa 6.2 indican que todos los RNGs a excepción del Chua-Saturada presentan buenos resultados y que en la mayoría la diferencia en los porcentajes de cobertura no se ve una diferencia radical cuando se aumenta la frecuencia de la señal rápida. Sin embargo nuevamente pensando en una futura implementación en hardware el RNG Saturada-Chua resulta der el más eficiente.

RNG / DATOS	Chua-Lorenz	Chua-Saturada	Lorenz-Chua	Lorenz-Saturada	Saturada-Chua	Saturada-Lorenz
Porcentaje en los Puntos de Equilibrio	subida	subida	subida	bidireccional	bidireccional	bidireccional
Secciones óptimas	1-1	1-1	1-1000	1-1	1-1000	1-10
Porcentaje en las secciones óptimas	80.6667%	46.0000%	82.0000%	80.1111%	81.5556%	81.0000%
No. de iteraciones para generar 1500 trayectorias	6368851	7434254	15879188	14619488	17983309	17621215
Bits para generar 1500 trayectorias en las secciones óptimas	16990	19840	17230	15880	17930	17550
Bits descartados por estar fuera del llimite	1990	4840	2230	880	2930	2550
Pruebas NIST exitosas con 10,000 bits	6/8	3/8	6/8	5/8	6/8	6/8
Pruebas NIST exitosas con 100,000 bits	5/9	4/9	5/9	5/9	6/9	6/9

Figura 6.2: Tabla comparativa de los RNGs 2 señales a diferente frecuencia.

6.3. RNGs híbrido

En el caso del RNG hibrido combinando los sistemas de Chua-Lorenz-Saturada no denota un cambio significativo o radical en los porcentajes de cobertura, a pesar de lo que se creía en un principio si fue necesario implementar técnicas de post-procesamiento, además resulta complicado implementar operaciones para que los sistemas no se desestabilicen al momento de la conmutación entre ellos. Otra razón es que, pensando una implementación en hardware, no resulta conveniente implementar tres sistemas para obtener resultados similares a los que nos proporcionan los RNGs anteriores.

6.4. RNGs Max. Exp. de Lyapunov

En la tercer sección se hizo un análisis para determinar la dependencia del máximo exponente de Lyapunov en los RNGs basados en los sistemas de Chua normalizado y el sistema basado en una función saturada. El análisis se hizo con tres valores diferentes en los parámetros de control del sistema para aumentar el máximo exponente de Lyapunov, lo que dio como resultado 6 RNGs para analizar.

Se pudo comprobar que sí existe una pequeña correlación entre el cambio del exponente de Lyapunov y los porcentajes de cobertura, la cual se ve más pronunciada en los RNGs de los sistemas basados en una función saturada. Cuando el sistema posee un máximo exponente de Lyapunov mayor, el porcentaje de cobertura promedio aumenta y, como se puede observar en las graficas de superficie, se hace más uniforme. En el caso del sistema de Chua se ve una mejoría significtiva en solo uno de los casos en que el máximo exponente de Lyapunov fue optimizado, en el otro caso el porcentaje promedio fue muy similar al sistema original.

RNG / DATOS	Saturada 2x1 (L1)	Saturada 2x1 (L2)	Saturada 2x1 (L3)	Chua Norm. (L1)	Chua Norm. (L2)	Chua Norm. (L3)
Porcentaje en los Puntos de Equilibrio	66.1111%	53.1111 %	27.8889 %	51.5556 %	73.7778 %	32.6667 %
Secciones óptimas	c1=3.2 c2=-4.4	c1=1.6 c2=-3.8	c1=3.9 c2=-2.6	c1=4.1 c2=-4.4	c1=1.3 c2=-1.4	c1=4.1 c2=-1.1
Porcentaje en las secciones óptimas	83.2222%	82.7778%	83.6667%	82.8889 %	83.0000 %	81.8889 %
No. de iteraciones para generar 1500 trayectorias	55177100	24242747	106677015	462492823	116289058	131136013
Porcentajes por encima del 60%	883	1314	935	260	1049	458
Porcentajes por encima del 70%	603	875	760	115	700	249
Porcentajes por encima del 80%	143	146	246	15	97	16
Porcentaje promedio	57.4542%	69.1299%	73.2444%	48.3624 %	58.7167%	48.8854%
Bits para generar 1500 trayectorias en las secciones óptimas	60548	50270	126086	66850	46004	49998
Bits descartados por la técnica VN	25208	17430	90886	32030	12504	13798
Bits descartados por estar fuera del llimite	2670	1420	2600	2410	1750	3100
Pruebas NIST exitosas con 10,000 bits	6/8	6/8	5/8	5/8	5/8	6/8
Pruebas NIST exitosas con 100,000 bits	6/9	6/9	6/9	6/9	6/9	7/9
Máximo exponente de Lyapunov	0.0011	0.0018	0.0031	0.00021537	0.000273	0.00038638

Figura 6.3: Tabla comparativa de los RNGs (Máximo exponente de Lyapunov).

Finalmente se realizaron las pruebas estadísticas NIST en todos los RNGs diseñados. En algunos RNGs no fue posible hacer la prueba con un millón de bits debido a la gran cantidad de iteraciones que se necesitan para obtenerlos, sin embargo la prueba se realizó en todos los casos para 10,000 y 100,000 bits.

Los RNGs pasaron la mayoría de las pruebas de aleatoriedad, sin embargo como se puede apreciar en las tablas, conforme se va incrementando el número de bits en la prueba NIST, las probabilidades de pasar la prueba se vuelven cada vez menores, esto se debe a que el paquete

NIST fue diseñado para evaluar RNGs y PRNGs con fines criptográficos.

Capítulo 7

Conclusiones

Después de estudiar y simular los sistemas caóticos de Chua, Lorenz y el sistema basado en una función saturada, se propuso una técnica para generar rutas de exploración mediante generadores de números aleatorios, utilizando los sistemas caóticos antes mencionados como fuente de entropía. En cada RNG diseñado se realizó un fuerte estudio para determinar las secciones de Poincaré óptimas en donde se producían porcentajes de cobertura altos y se comprobó la importancia de implementar técnicas de post-procesamiento (técnica de Von Neumman) en la generación de números aleatorios. Después de comparar los porcentajes de todos los RNGs se observó que el RNG basado en el sistema de Lorenz presentó una distribución más uniforme y porcentajes de cobertura más altos que los demás.

En este trabajo se diseñaron un total de 23 RNGs. Partiendo del hecho de que un RNG haya presentado un porcentaje de cobertura mayor en las "secciones de Poincaré óptimas" que el otro no garantiza que sea mas eficiente. Esto es, el hecho de que un RNG cubra el 100 % del área NO garantiza que las trayectorias sean aleatorias. Asimismo concluye que cualquier sección de Poincaré que se encuentre cerca o por encima del 80 % se puede considerar como una combinación óptima para la obtención de bits aleatorios.

Una forma de concluir que un RNG es más eficiente que el otro sería por medio del análisis de los porcentaje promedio y por medio de las gráficas de superficie (uniformidad). El sistema de Lorenz presentó un porcentaje de cobertura promedio mayor, más uniformidad que los demás y un número mayor de secciones con porcentajes que rondan o superan el 80 % de cobertura.

Por otro lado, existe algo que irremediablemente se tiene que considerar: la implementación en hardware. Es por eso que el RNG basado en un sistema caótico PWL resulta ser el más conveniente, presentando porcentajes de cobertura que superan el 80 % .

Es importante destacar el ancho de paso como un parámetro importante en el análisis, de acuerdo a las con las simulaciones realizadas.

Por otra parte, en el RNG dual se aprovechó la dinámica compleja que proporcionan las señales $x(t)$ y $y(t)$ del sistema basado en una serie de funciones saturadas de 2×2. Con ello se obtuvieron dos fuentes para generar bits aleatorios en un solo sistema.

En lo que respecta a los RNGs que combinan dos sistemas caóticos a diferente frecuencia se observaron porcentajes de cobertura altos y con la ventaja de prescindir de técnicas de post-procesamiento, lo que traduce en menos tiempo de cómputo. Se propuso una técnica para la generación de sistemas híbridos combinando tres sistemas caóticos en el diseño de un RNG. En

el caso del RNG hibrido no se encontró una mejora radical en los porcentajes de cobertura con respecto a los demás, a pesar de lo que se creía en un principio si fue necesario implementar técnicas de post-procesamiento y resulta complicado implementar operaciones para que los sistemas no se desestabilicen al momento de la conmutación entre ellos, además de que el tiempo de cómputo se incrementa considerablemente.

Se diseñaron seis RNGs con 2 señales caóticas a diferente frecuencia, del análisis de estos se puede concluir que: no se implementó la técnica de Von Neumann ya que la mayoría presentaron porcentajes de cobertura altos sin necesidad de implementar un post-procesamiento (mayor del 80 %), los resultados obtenidos indican que todos los RNGs, a excepción del Chua-Saturada, presentaron porcentajes de cobertura altos, en la mayoría de los RNGs la diferencia en los porcentajes de cobertura no presentó gran diferencia cuando se aumentó la frecuencia de la señal rápida. El RNG Saturada-Chua resulta ser el más conveniente de acuerdo a los criterios antes señalados.

En el caso del RNG hibrido (Chua-Lorenz-Saturada) se puede concluir: no denotó un cambio significativo o radical en los porcentajes de cobertura, fue necesario implementar técnicas de post-procesamiento, resulta complicado implementar operaciones para que los sistemas no se desestabilicen al momento de la conmutación. Finalmente, no resulta conveniente combinar la dinámica de tres sistemas para obtener resultados similares a los que nos proporcionan los RNGs con un solo sistema caótico.

Además, se hizo un análisis para determinar la dependencia del máximo exponente de Lyapunov en los RNGs basados en los sistemas de Chua normalizado y el sistema basado en una función saturada. Se pudo comprobar que sí existe una correlación entre el aumento del máximo exponente de Lyapunov y el aumento en los porcentajes de cobertura, la cual se ve más pronunciada en los RNGs de los sistemas basados en una función saturada.

Finalmente la mayoría de las pruebas estadísticas del paquete NIST fueron exitosas, y a pesar de que los RNGs no cumplen con todas las normas tan estrictas que demanda un RNG con fines criptográficos se puede concluir que los RNGs cumplen de forma satisfactoria el objetivo para los que fueron diseñados: planear rutas de exploración de forma aleatoria e impredecible. Además de lo enriquecedor que resultó el estudio de la dinámica caótica en el proceso de diseño de los RNGs. Se realizaron las pruebas estadísticas NIST en todos los RNGs diseñados. La prueba se realizó en todos los casos para 10,000 y 100,000 bits. Los RNGs pasaron la mayoría de las pruebas de aleatoriedad, obviamente salvo aquellos en los que el porcentaje de cobertura en las secciones óptimas era demasiado bajo. Sin embargo, como se puede apreciar en los resultados presentados, casi todos los RNGs pasaron la misma cantidad de pruebas. Se concluye entonces que los RNGs basados en caos que se diseñaron resultan ser una buena herramienta en la generación de rutas aleatorias, además de lo enriquecedor que resultó analizar los RNGs desde todos los puntos de vista que se abordaron: en función de secciones de Poincaré, escalamiento en frecuencias y combinación de sistemas caóticos (en la búsqueda de estrategias para no implementar técnicas de post-procesamiento), en la dependencia y el efecto del Máximo exponente de Lyapunov y pruebas estadísticas.

Apéndice A

Versiones alternas de RNGs con un sistema caótico

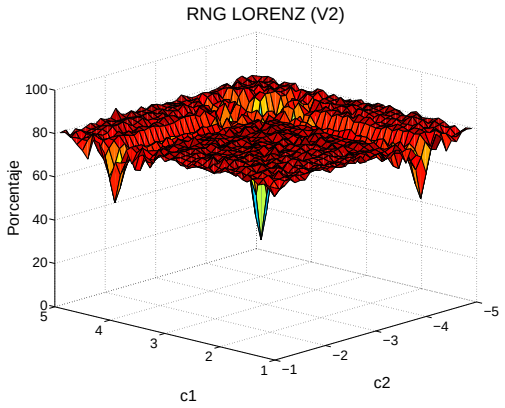
En este Apéndice se muestran otras versiones de los RNGs basados en un sistema caótico. El objetivo es averiguar si existe un cambio significativo en los porcentajes de cobertura de los RNGs cuando los parámetros del sistema son modificados. También, resulta interesante comparar RNGs diseñados con el mismo sistema pero con diferente ancho de paso de integración para comprobar la importancia de simular los sistemas con un ancho de paso optimizado.

A.1. RNG basado en el sistema de Lorenz con $\gamma = 28$

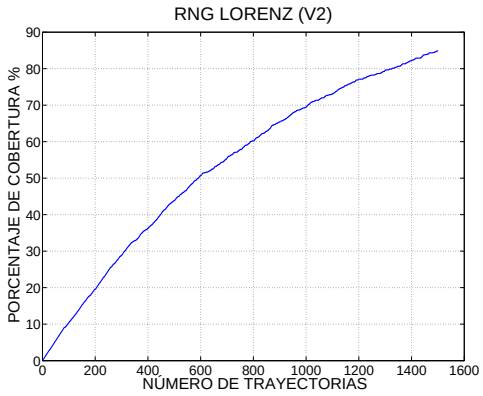
El RNG que se presenta a continuación, al igual que el RNG presentado en la sección 4.1.3 utiliza la señal $x(t)$ del sistema de Lorenz como fuente de entropía. La diferencia radica en que el parámetro γ (principal parámetro de control del sistema) fue modificado para observar si existe algún cambio significativo en los porcentajes de cobertura. Las secciones de Poincaré en donde se encontraron los porcentajes de cobertura más altos se ubican en $c1 = 4.9$ y $c2 = -2.1$ con 84.8888 %. En la Tabla A.1 se presentan los resultados completos.

Datos	**Resultados**
Ancho de paso	0.0037
Porcentaje de cobertura en los puntos de equilibrio	80.7778 %
Secciones óptimas	$c1 = 4.9$ $c2 = -2.1$
Porcentaje de cobertura en las secciones óptimas	84.8888 %
Número de iteraciones para generar 1500 trayectorias en las secciones óptimas	137729533
Porcentajes mayores a 60 %	1579
Porcentajes mayores a 70 %	1507
Porcentajes mayores a 80 %	823
Porcentaje promedio	78.5033 %
Trayectorias necesarias para cubrir el 100 % del área	4793
Iteraciones necesarias para cubrir el 100 % del área	445672051
Total de bits generados en las secciones óptimas	371088
Bits descartados por la técnica de Von Neumann	337408
Bits descartados por estar fuera del límite ($1 \leq x \leq 30$)	1840
Máximo exponente de Lyapunov	0.0032

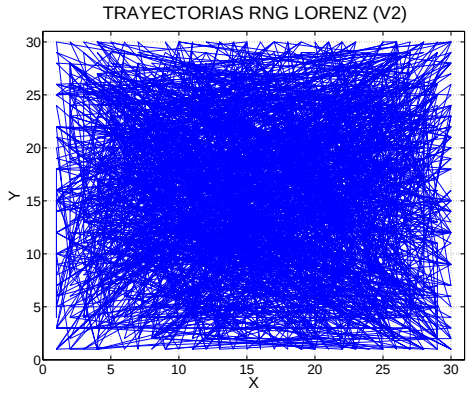
Cuadro A.1: Resultados obtenidos del RGN basado en el sistema de Lorenz ($\gamma = 28$).



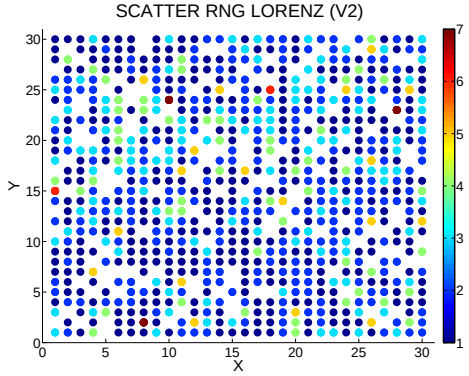
(a) Gráfica de porcentajes



(b) Porc. de cobertura en $c1 = 4.9$ y $c2 = -2.1$



(c) Trayectorias RNG Lorenz



(d) Gráfica de dispersión

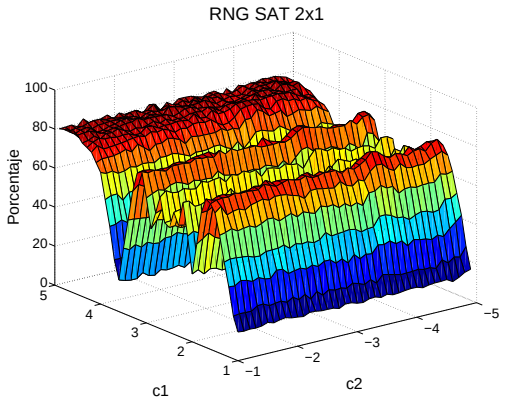
Figura A.1: RNG basado en el sistema de Lorenz ($\gamma = 28$).

A.2. RNG del sistema de la función saturada de 2x1

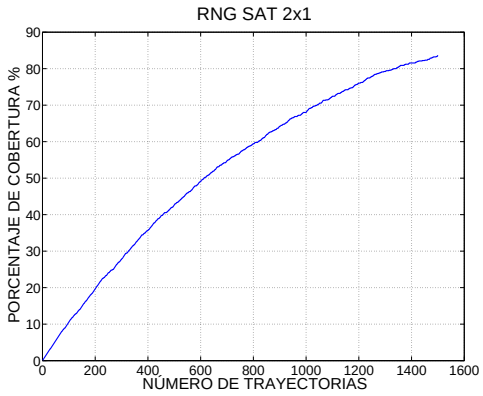
En este RNG se utiliza la señal $x(t)$ del sistema basado en una función saturada como fuente de entropía.

Datos	**Resultados**
Ancho de paso	0.0118
Porcentaje de cobertura en los puntos de equilibrio	54.0000 %
Secciones óptimas	$c1 = 4.8$ $c2 = -2.5$
Porcentaje de cobertura en las secciones óptimas	83.5555 %
Número de iteraciones para generar 1500 trayectorias en las secciones óptimas	123962019
Porcentajes mayores a 60 %	881
Porcentajes mayores a 70 %	568
Porcentajes mayores a 80 %	181
Porcentaje promedio	59.4619 %
Trayectorias necesarias para cubrir el 100 % del área	7002
Iteraciones necesarias para cubrir el 100 % del área	580255075
Total de bits generados en las secciones óptimas	28099528
Bits descartados por la técnica de Von Neumann	28065648
Bits descartados por estar fuera del límite ($1 \leq x \leq 30$)	1940
Máximo exponente de Lyapunov	0.00082

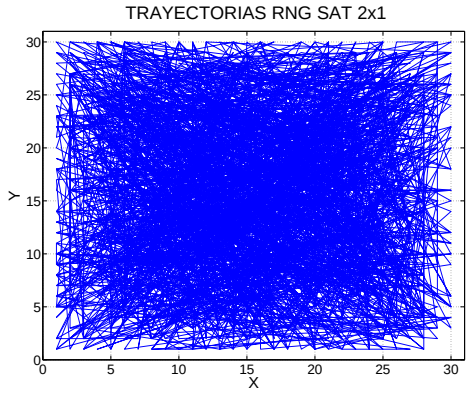
Cuadro A.2: Resultados obtenidos del RGN del sistema basado en una función saturada 2×1.



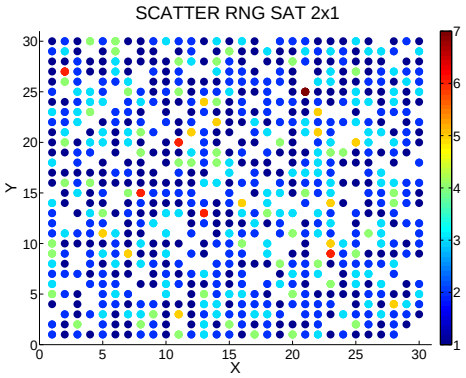
(a) Gráfica de porcentajes



(b) Porc. de cobertura en $c1 = 4.8$ y $c2 = -2.5$



(c) Trayectorias RNG Saturada 2x1



(d) Gráfica de dispersión

Figura A.2: RNG del sistema basado en una función saturada 2×1.

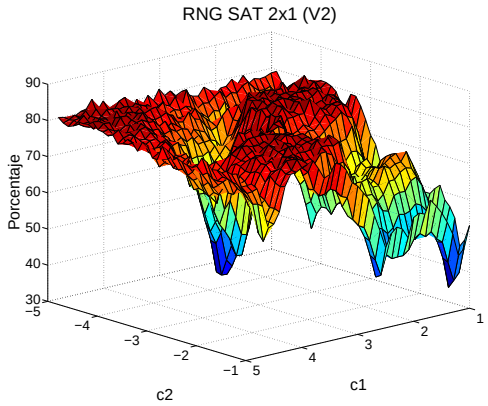
Dicho sistema está basado en una función PWL con una pendiente de 9, un ancho de paso óptimo en función de los eigenvalores del sistema y con dos niveles de saturación de 2.5 unidades. En las secciones $c1 = 4.8$ y $c2 = -2.5$ se obtiene el porcentaje de cobertura más alto con 83.5555 %. En la Tabla A.2 se presentan los datos más importantes.

A.3. RNG del sistema basado en una función saturada de 2x1 (V2)

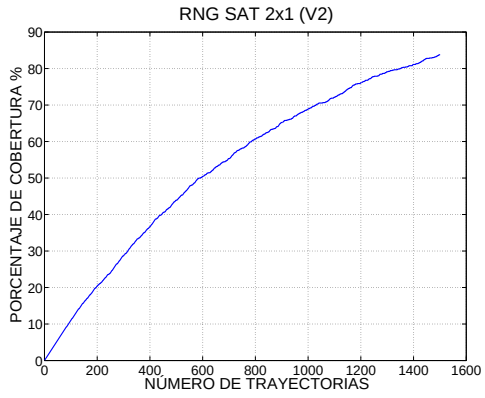
En este RNG se utiliza la señal $x(t)$ del sistema basado en una función saturada como fuente de entropía. La diferencia con el RNG anterior radica en que ahora la pendiente de la función PWL tiene un valor de 100. Además, se modificó el ancho de paso de integración a 0.15. Las secciones de Poincaré con el porcentaje de cobertura más alto se ubican en $c1 = 4.3$ y $c2 = -4.9$ con 83.7777 %. En la Tabla A.3 se presentan los datos más importantes.

Datos	**Resultados**
Ancho de paso	0.15
Porcentaje de cobertura en los puntos de equilibrio	63.7778 %
Secciones óptimas	$c1 = 4.3$ $c2 = -4.9$
Porcentaje de cobertura en las secciones óptimas	83.7777 %
Número de iteraciones para generar 1500 trayectorias	45986500
Porcentajes mayores a 60 %	1386
Porcentajes mayores a 70 %	1084
Porcentajes mayores a 80 %	325
Porcentaje promedio	71.972 %
Trayectorias necesarias para cubrir el 100 % del área	17960
Iteraciones necesarias para cubrir el 100 % del área	90602527
Total de bits generados en las secciones óptimas	104872
Bits descartados por la técnica de Von Neumann	71412
Bits descartados por estar fuera del límite ($1 \leq x \leq 30$)	1730
Máximo exponente de Lyapunov	0.0209

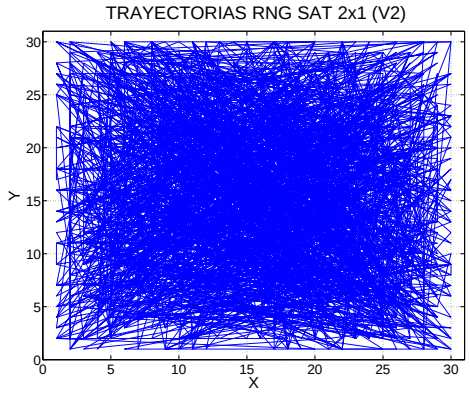
Cuadro A.3: Resultados obtenidos del RGN del sistema basado en una función saturada 2×1 *(V2).*



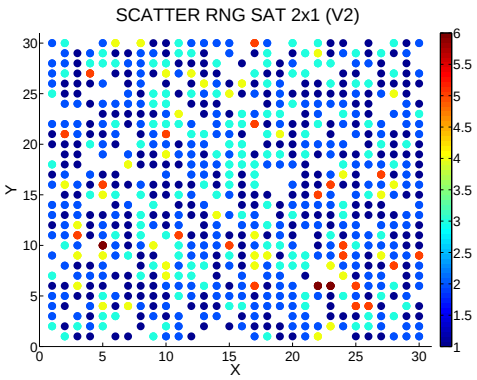
(a) Gráfica de porcentajes.



(b) Porc. de cobertura en $c1 = 4.3$ y $c2 = -4.9$.



(c) Trayectorias RNG Saturada 2x1.



(d) Gráfica de dispersión.

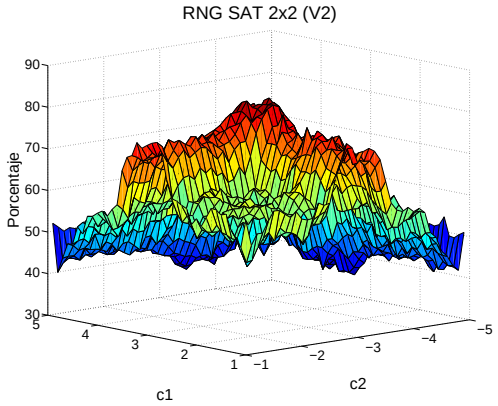
Figura A.3: RNG basado en una función saturada 2×1 *(V2).*

A.4. RNG del sistema de una función PWL de 2x2 (V2)

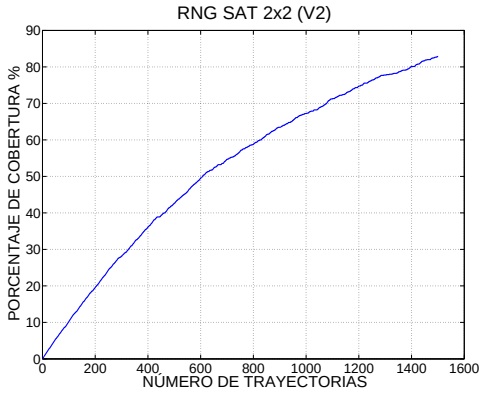
En este RNG tambien utiliza la señal $x(t)$ del sistema basado en una función saturada 2x2 como fuente de entropía, la diferencia con el RNG de la sección 4.1.6 consiste en aumentar el ancho de paso de integración, quedando definido en $h = 0.15$. Las secciones $c1 = 2.9$ y $c2 = -2.7$ presentaron el porcentaje de cobertura mayor con un 82.8888 %.

Datos	**Resultados**
Ancho de paso	0.15
Porcentaje de cobertura en los puntos de equilibrio	57.8889 %
Secciones óptimas	$c1 = 2.9$ $c2 = -2.7$
Porcentaje de cobertura en las secciones óptimas	82.8888 %
Número de iteraciones para generar 1500 trayectorias en las secciones óptimas	2601018
Porcentajes mayores a 60 %	736
Porcentajes mayores a 70 %	313
Porcentajes mayores a 80 %	25
Porcentaje promedio	58.7167 %
Trayectorias necesarias para cubrir el 100 % del área	40559
Iteraciones necesarias para cubrir el 100 % del área	62215562
Total de bits generados en las secciones óptimas	48776
Bits descartados por la técnica de Von Neumann	14436
Bits descartados por estar fuera del límite $(1 \leq x \leq 30)$	2170
Máximo exponente de Lyapunov	0.0187

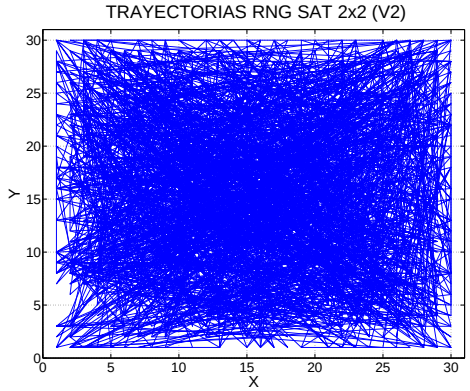
Cuadro A.4: Resultados del RGN basado en el sistema basado en una función saturada 2×2 *(V2).*



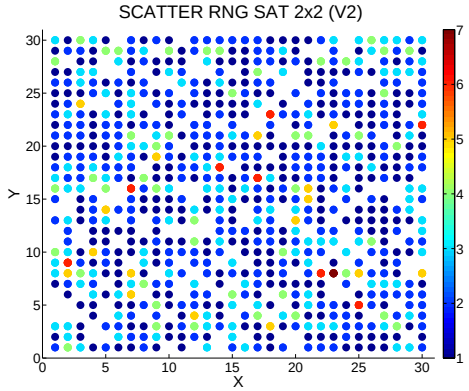
(a) Gráfica de porcentajes.



(b) Porc. de cobertura en $c1 = 2.9$ y $c2 = -2.7$.



(c) Trayectorias RNG Saturada 2x2



(d) Gráfica de dispersión

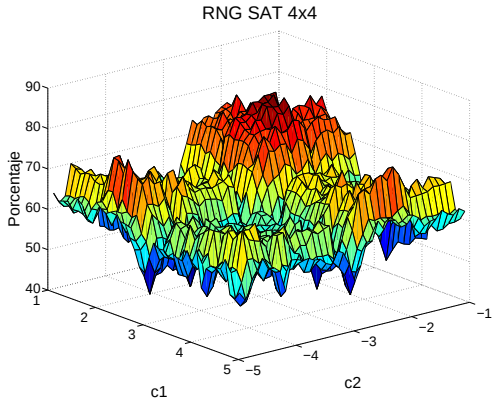
Figura A.4: RNG del sistema basado en una función saturada 2x2 (V2).

A.5. RNG del sistema basado en una función PWL (4x4)

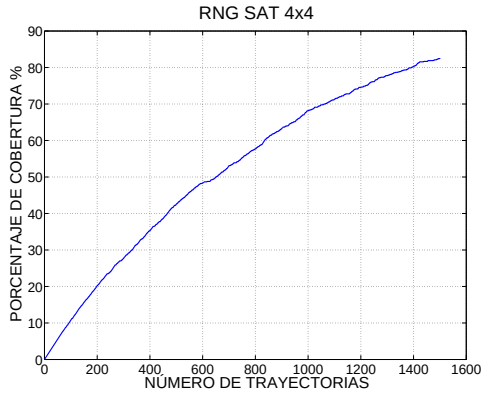
En este RNG se utiliza la señal $x(t)$ del sistema basado en una función saturada de cuatro enrollamientos en dos dimensiones (4x4). Las secciones de Poincaré donde se obtiene el porcentaje de cobertura más alto se encuentra en $c1 = 1.7$ y $c2 = -1.8$ con 84.1111 %.

Datos	**Resultados**
Ancho de paso	0.0158
Porcentaje de cobertura en los puntos de equilibrio	65.2222 %, 72.3333 %, 58.2222 %
Secciones óptimas	$c1 = 2.1$ $c2 = -2$
Porcentaje de cobertura en las secciones óptimas	82.4444 %
Número de iteraciones para generar 1500 trayectorias	102238110
Porcentajes mayores a 60 %	1099
Porcentajes mayores a 70 %	227
Porcentajes mayores a 80 %	15
Porcentaje promedio	62.976 %
Trayectorias necesarias para cubrir el 100 % del área	9831
Iteraciones necesarias para cubrir el 100 % del área	674764627
Total de bits generados en las secciones óptimas	82578
Bits descartados por la técnica de Von Neumann	49538
Bits descartados por estar fuera del límite ($1 \leq x \leq 30$)	1520
Máximo exponente de Lyapunov	0.0015

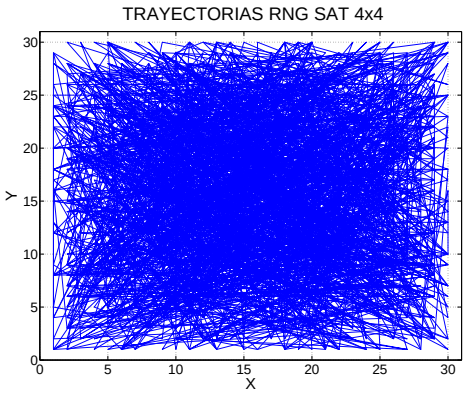
Cuadro A.5: Resultados del RGN basado en el sistema basado en una función saturada 4×4.



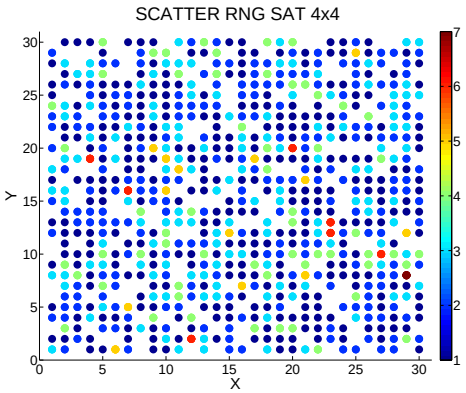
(a) Gráfica de porcentajes de todas las secciones



(b) Porcentaje de cobertura en las secciones óptimas



(c) Trayectorias RNG Saturada 4x4



(d) Gráfica de dispersión

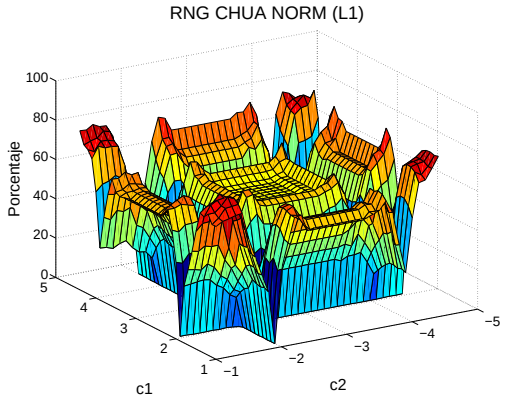
Figura A.5: RNG del sistema basado en una función saturada 4×4.

A.6. RNG Chua Normalizado (L1)

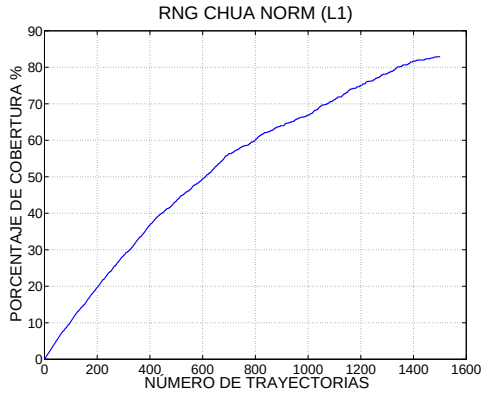
Como se mencionó anteriormente, para llevar a cabo el análisis de la dependencia del máximo exponente de Lyapunov se realizan tres versiones del sistema basado en el sistema de Chua normalizado.

Para ello se definen los parametros de acuerdo a la Tabla 4.1. Para aproximar los valores correspondientes a cada exponente de Lyapunov los máximos exponentes de Lyapunov se utilizó la herramienta Tisean 3.0.0 y MATLAB.

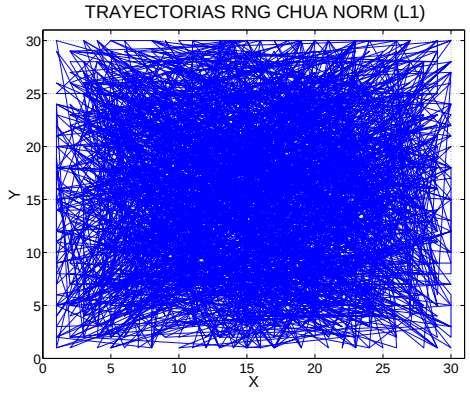
En la gráfica de la Figura A.6(a) se observan los porcentajes correspondientes a todas las secciones de Poincaré, en donde $c1 = 4.1$ y $c2 = -4.4$ resultan ser las secciones en donde se obtiene el porcentaje de cobertura más alto con valor de 82.8889 %.



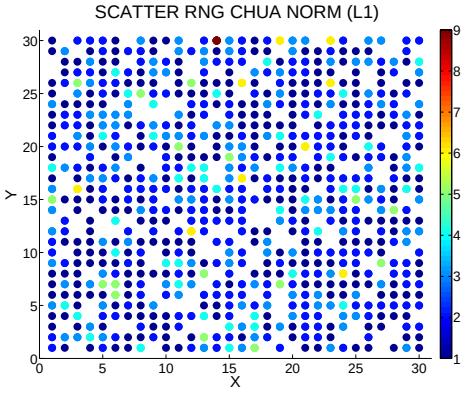
(a) Gráfica de porcentajes.



(b) Porc. de cobertura en $c1 = 4.1$ y $c2 = -4.4$.



(c) Trayectorias de exploración.



(d) Gráfica de dispersión.

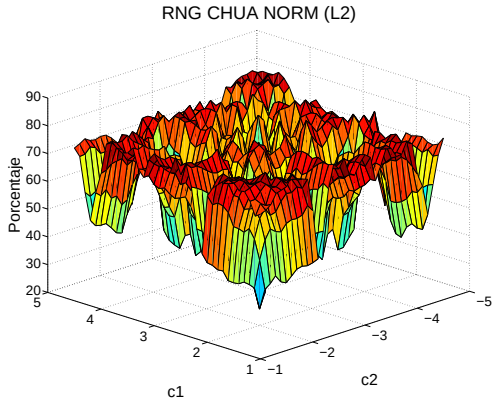
Figura A.6: RNG basado en el circuito de Chua normalizado (L1).

Datos	**Resultados**
Ancho de paso	0.001
Porcentaje de cobertura en los puntos de equilibrio	51.5556 %
Secciones óptimas	$c1 = 4.1$ $c2 = -4.4$
Porcentaje de cobertura en las secciones óptimas	82.8889 %
Número de iteraciones para generar 1500 trayectorias en las secciones óptimas	462492823
Porcentajes mayores a 60 %	260
Porcentajes mayores a 70 %	115
Porcentajes mayores a 80 %	15
Porcentaje promedio	48.3624 %
Trayectorias necesarias para cubrir el 100 % del área	6104
Iteraciones necesarias para cubrir el 100 % del área	1876008021
Total de bits generados en las secciones óptimas	66850
Bits descartados por la técnica de Von Neumann	32030
Bits descartados por estar fuera del límite ($1 \leq x \leq 30$)	2410
Máximo exponente de Lyapunov	0.00021537

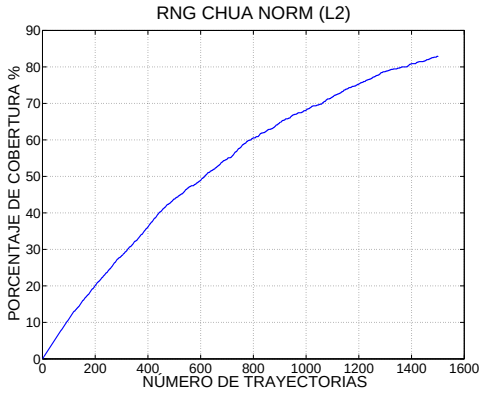
Cuadro A.6: Resultados obtenidos del RGN del sistema de Chua normalizado (L1).

A.6.1. RNG Chua normalizado (L2)

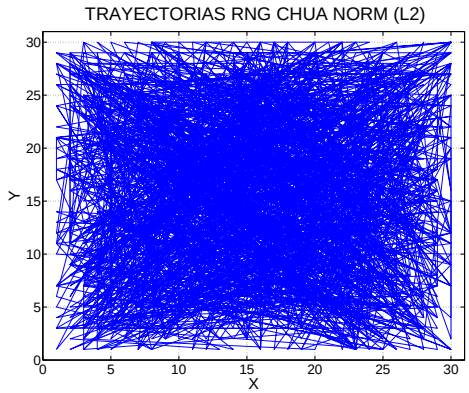
En este RNG se utiliza la señal $x(t)$ del sistema basado en el sistema de Chua normalizado con los parámetros $\alpha = 16.57$ y $\beta = 25.88$.



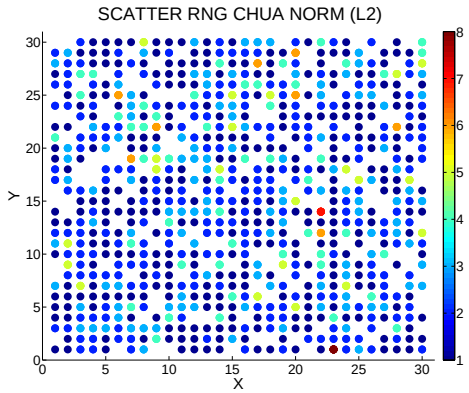
(a) Gráfica de porcentajes



(b) Porc. de cobertura en $c1 = 1.3$ y $c2 = -1.4$.



(c) Trayectorias de exploración



(d) Gráfica de dispersión

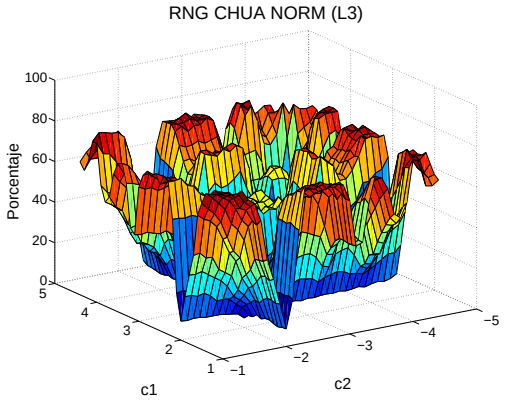
Figura A.7: RNG basado en el circuito de Chua normalizado (L2).

En las secciones $c1 = 1.3$ y $c2 = -1.4$ se obtiene el porcentaje de cobertura más alto con 83 %. En la Tabla A.7 se presentan los datos más relevantes.

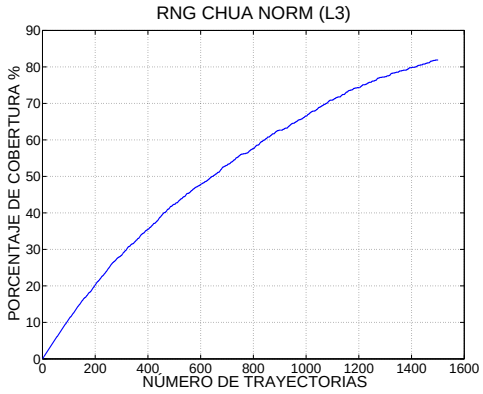
Datos	**Resultados**
Ancho de paso	0.001
Porcentaje de cobertura en los puntos de equilibrio	73.7778 %
Secciones óptimas	$c1 = 1.3$ $c2 = -1.4$
Porcentaje de cobertura en las secciones óptimas	83.0000 %
Número de iteraciones para generar 1500 trayectorias en las secciones óptimas	116289058
Porcentajes mayores a 60 %	1049
Porcentajes mayores a 70 %	700
Porcentajes mayores a 80 %	97
Porcentaje promedio	68.3748 %
Trayectorias necesarias para cubrir el 100 % del área	8010
Iteraciones necesarias para cubrir el 100 % del área	622598260
Total de bits generados en las secciones óptimas	46004
Bits descartados por la técnica de Von Neumann	12504
Bits descartados por estar fuera del límite ($1 \leq x \leq 30$)	1750
Máximo exponente de Lyapunov	0.000273

Cuadro A.7: Resultados obtenidos del RGN del sistema de Chua normalizado (L2).

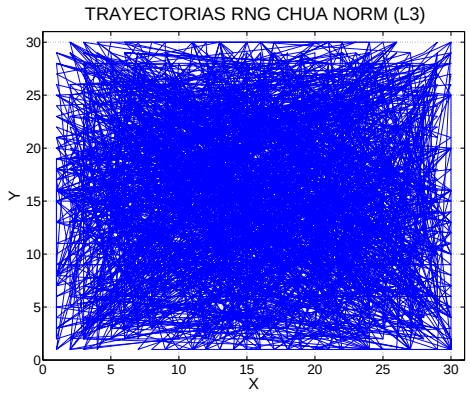
A.6.2. RNG Chua normalizado (L3)



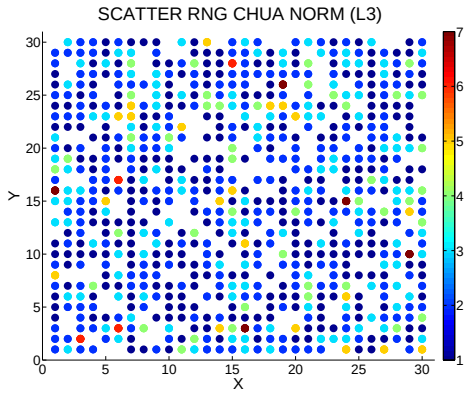
(a) Gráfica de porcentajes.



(b) Porc. de cobertura en $c1 = 1.3$ y $c2 = -1.4$.



(c) Trayectorias de exploración



(d) Gráfica de dispersión

Figura A.8: RNG del sistema de Chua normalizado (L3).

En las secciones de Poincaré situadas en $c1 = 4.1$ y $c2 = -1.1$ se obtiene el porcentaje de cobertura más alto con 81.8889 %. En este RNG se utiliza la señal $x(t)$ del sistema basado en el sistema de Chua normalizado con los parámetros $\alpha = 17.96$ y $\beta = 31.66$.

Datos	**Resultados**
Ancho de paso	0.001
Porcentaje de cobertura en los puntos de equilibrio	32.6667 %
Secciones óptimas	$c1 = 4.1$ $c2 = -1.1$
Porcentaje de cobertura en las secciones óptimas	81.8889 %
Número de iteraciones para generar 1500 trayectorias en las secciones óptimas	131136013
Porcentajes mayores a 60 %	458
Porcentajes mayores a 70 %	249
Porcentajes mayores a 80 %	16
Porcentaje promedio	48.8854 %
Trayectorias necesarias para cubrir el 100 % del área	6989
Iteraciones necesarias para cubrir el 100 % del área	625647763
Total de bits generados en las secciones óptimas	49998
Bits descartados por la técnica de Von Neumann	13798
Bits descartados por estar fuera del límite ($1 \leq x \leq 30$)	3100
Máximo exponente de Lyapunov	0.00038638

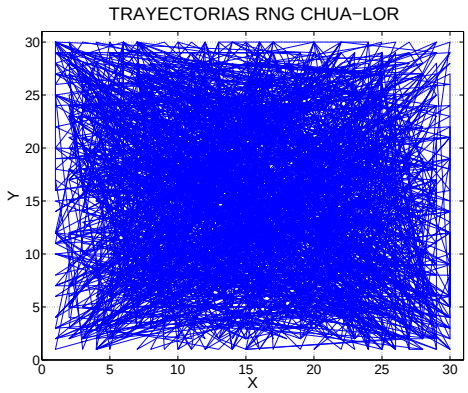
Cuadro A.8: Resultados obtenidos del RGN basado en el sistema de Chua normalizado (L3).

Apéndice B

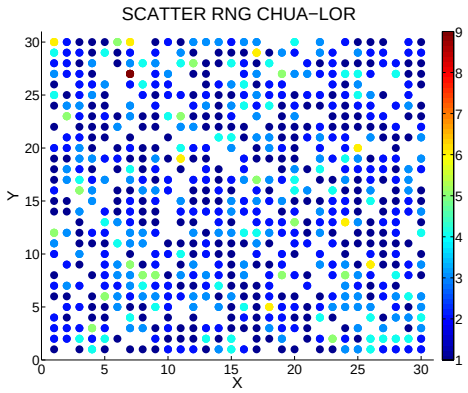
Gráficas y resultados de los RNGs combinando dos sistemas caóticos a diferente frecuencia

En este Apéndice se presentan los resultados de los RNGs diseñados a partir de dos sistemas caóticos a diferente frecuencia, así como las gráficas de porcentaje de cobertura, trayectorias planeadas y coordenadas visitadas en el área de exploración en los FS óptimos definidos en la sección 4.2. Dichos resultados permiten evaluar y comparar la eficiencia entre cada RNG.

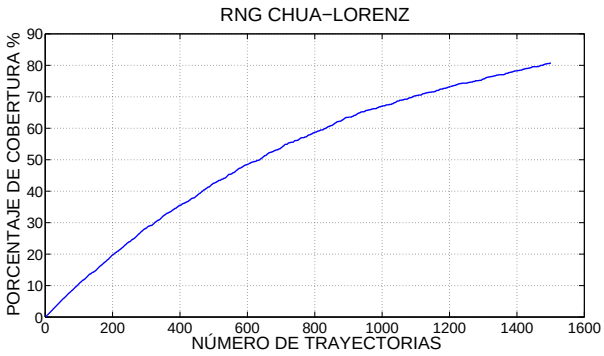
B.1. RNG Chua-Lorenz



(a) Trayectorias planeadas



(b) Gráfica de dispersión



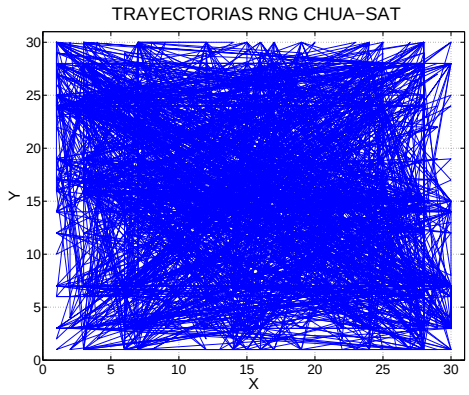
(c) Porcentaje de cobertura (FS óptimo)

Figura B.1: RNG Chua-Lorenz.

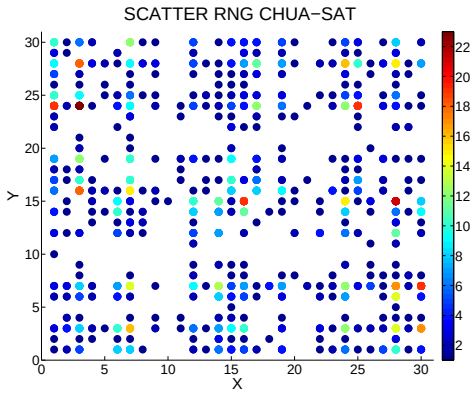
Datos	**Resultados**
Flanco	Subida
Factor de escalamiento de la frecuencia (FS)	1-1
Porcentaje de cobertura	80.6667 %
Número de iteraciones para generar 1500 trayectorias	6368851
Trayectorias necesarias para cubrir el 100 % del área	7890
Iteraciones necesarias para cubrir el 100 % del área	33739797
Total de bits generados	16990
Bits descartados por estar fuera del límite $(1 \leq x \leq 30)$	1990

Cuadro B.1: Resultados obtenidos del RGN Chua-Lorenz.

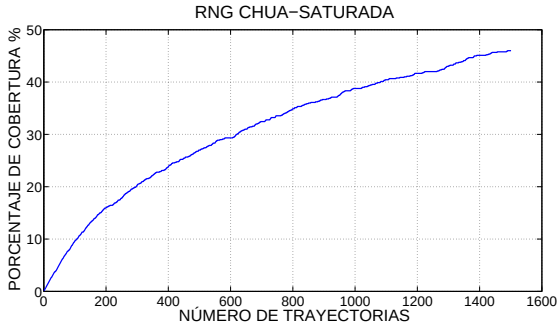
B.2. RNG Chua-Saturada



(a) Trayectorias de exploración.



(b) Gráfica de dispersión.



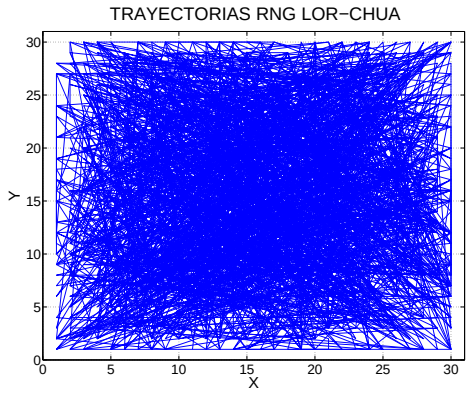
(c) Porcentaje de cobertura (FS óptimo).

Figura B.2: RNG Chua-Saturada.

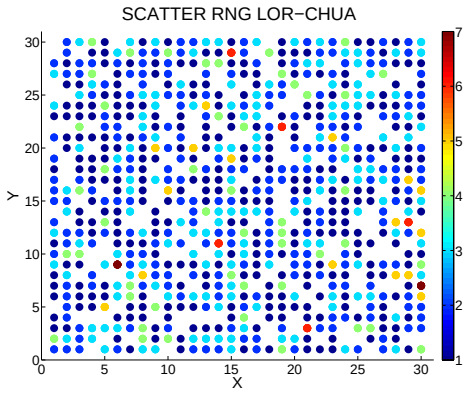
Datos	**Resultados**
Flanco	Subida
Factor de escalamiento de la frecuencia (FS)	1-1
Porcentaje de cobertura	46.0000 %
Número de iteraciones para generar 1500 trayectorias	7434254
Total de bits generados	19840
Bits descartados por estar fuera del límite ($1 \leq x \leq 30$)	4840

Cuadro B.2: Resultados obtenidos del RGN Chua-Saturada.

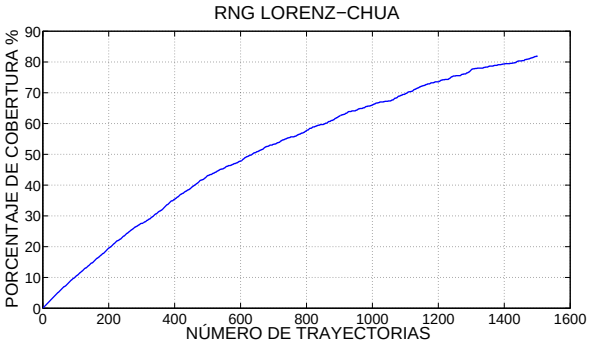
B.3. RNG Lorenz-Chua



(a) Trayectorias de exploración.



(b) Gráfica de dispersión.



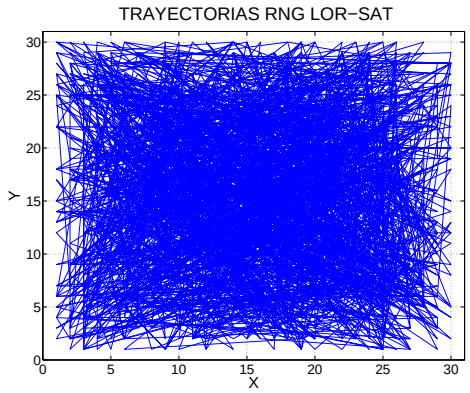
(c) Porcentaje de cobertura (FS óptimo).

Figura B.3: RNG Lorenz-Chua.

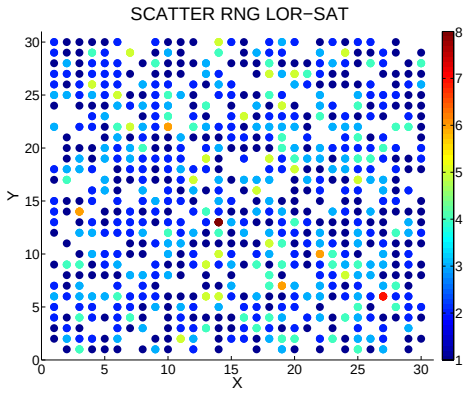
Datos	**Resultados**
Flanco	Subida
Factor de escalamiento de la frecuencia (FS)	F 1-1000
Porcentaje de cobertura	82.0000 %
Número de iteraciones para generar 1500 trayectorias	15879188
Trayectorias necesarias para cubrir el 100 % del área	6318
Iteraciones necesarias para cubrir el 100 % del área	67449762
Total de bits generados	17230
Bits descartados por estar fuera del límite $(1 \leq x \leq 30)$	2230

Cuadro B.3: Resultados obtenidos del RGN Lorenz-Chua.

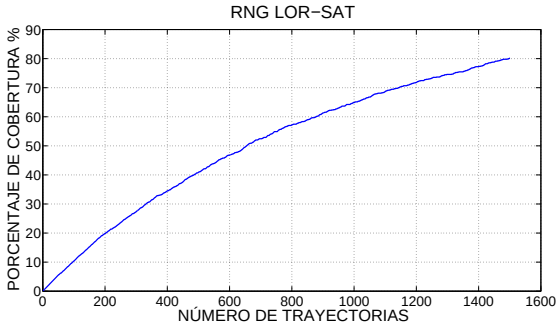
B.4. RNG Lorenz-Saturada



(a) Trayectorias de exploración.



(b) Gráfica de dispersión.



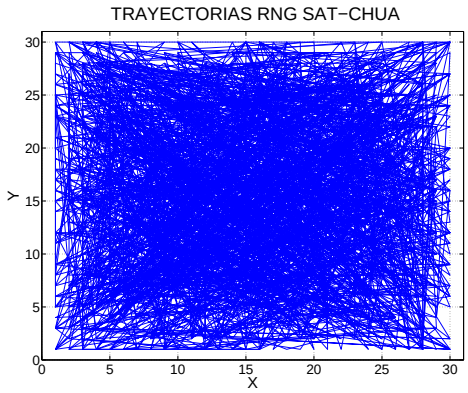
(c) Porcentaje de cobertura (FS óptimo).

Figura B.4: RNG Lorenz-Saturada.

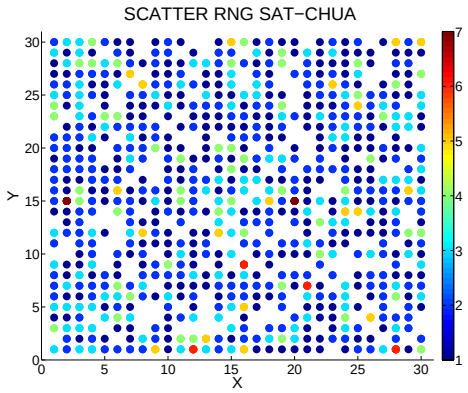
Datos	**Resultados**
Flanco	Subida
Factor de escalamiento de la frecuencia (FS)	F 1-1
Porcentaje de cobertura	80.1111 %
Número de iteraciones para generar 1500 trayectorias	14619488
Trayectorias necesarias para cubrir el 100 % del área	9258
Iteraciones necesarias para cubrir el 100 % del área	91382634
Total de bits generados	15880
Bits descartados por estar fuera del límite ($1 \leq x \leq 30$)	880

Cuadro B.4: Resultados obtenidos del RGN Lorenz-Saturada.

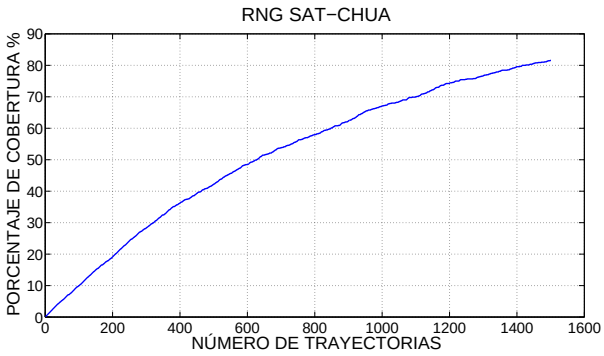
B.5. RNG Saturada-Chua



(a) Trayectorias de exploración.



(b) Gráfica de dispersión.



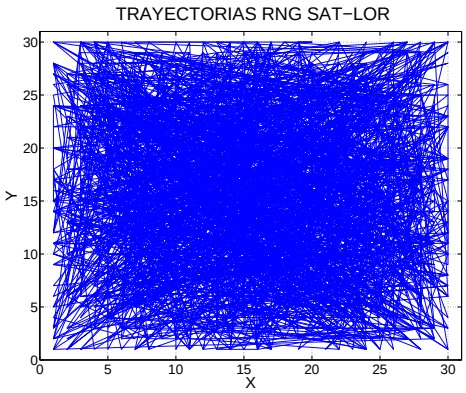
(c) Porcentaje de cobertura (FS óptimo).

Figura B.5: RNG Saturada-Chua.

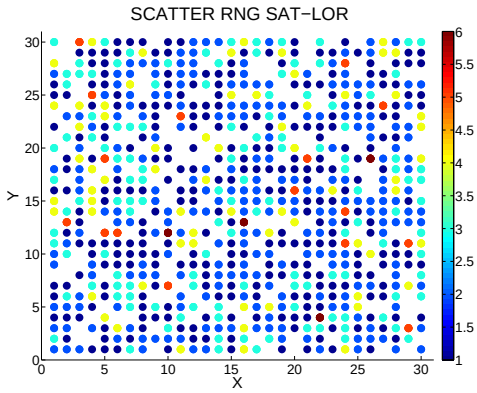
Datos	**Resultados**
Flanco	Bidireccional
Factor de escalamiento de la frecuencia (FS)	F 1-1000
Porcentaje de cobertura	81.5556 %
Número de iteraciones para generar 1500 trayectorias	17983309
Trayectorias necesarias para cubrir el 100 % del área	6938
Iteraciones necesarias para cubrir el 100 % del área	83086420
Total de bits generados	17930
Bits descartados por estar fuera del límite $(1 \leq x \leq 30)$	2930

Cuadro B.5: Resultados obtenidos del RGN Saturada-Chua.

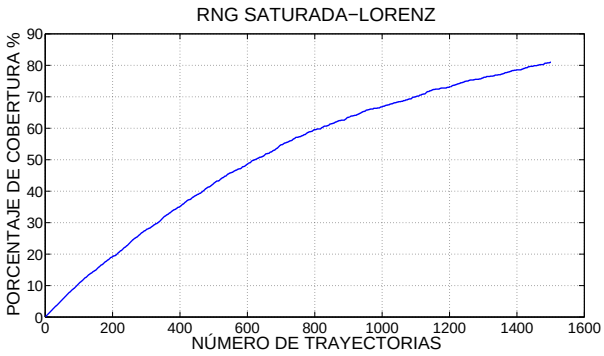
B.6. RNG Saturada-Lorenz



(a) Trayectorias de exploración.



(b) Gráfica de dispersión.



(c) Porcentaje de cobertura (FS óptimo).

Figura B.6: RNG Saturada-Lorenz.

Datos	**Resultados**
Flanco	Bidireccional
Factor de escalamiento de la frecuencia (FS)	F 1-10
Porcentaje de cobertura	81.0000 %
Número de iteraciones para generar 1500 trayectorias	17621215
Trayectorias necesarias para cubrir el 100 % del área	6385
Iteraciones necesarias para cubrir el 100 % del área	75278194
Total de bits generados	17550
Bits descartados por estar fuera del límite ($1 \leq x \leq 30$)	2550

Cuadro B.6: Resultados obtenidos del RGN Saturada-Lorenz.

Apéndice C

Exponente de Lyapunov

Este Apéndice se estudia el cálculo del máximo exponente de Lyapunov para series de tiempo, que es la herramienta utilizada en este trabajo de tesis para obtener el exponente positivo de Lyapunov de los sistemas antes estudiados.

C.1. Exponente de Lyapunov para series de tiempo

En un sistema dinámico la presencia de caos es determinada mediante el máximo exponente de Lyapunov. Los exponenetes de Lyapunov cuantifican la divergencia exponencial de trayectorias en las que sus condiciones iniciales son muy cercanas.

Una serie de tiempo es una secuencia de datos espaciados cronológicamente de forma uniforme, es decir un vector que contiene muestras de una señal evolucionando en el tiempo. Para series de tiempo producidas por sistemas dinámicos, la presencia de un exponente positivo es indicio de caos. Es por eso que en muchas aplicaciones es suficiente calcular solo el máximo exponente de Lyapunov (λ) para comprobar que existe caos.

El máximo exponente de Lyapunov se puede definir usando la ecuación (C.1), donde $d(t)$ es la distancia promedio en el tiempo t y C es una constante que normaliza la separación inicial.

$$d(t) = Ce^{\lambda_1 t}. \tag{C.1}$$

El primer paso para la aproximación implica reconstruir la dinámina del atractor a partir de una serie de tiempo, para ello se usa el método de retardos. La trayectoria reconstruida $\mathbf{X}$, puede ser expresada como una matriz donde cada fila es un vector en el espacio de fase. Esto es,

$$\mathbf{X} = \left(\mathbf{X}_1\ \mathbf{X}_2\ \ldots \mathbf{X}_M\right)^T, \tag{C.2}$$

donde $\mathbf{X}_i$ es el estado del sistema en el tiempo discreto i. Para un punto-N en la serie de tiempo, $\{x_1, x_2, \ldots, x_N\}$, cada $\mathbf{X}_i$ está dada por:

$$\mathbf{X}_i = \left(X_i\ X_{i+J}\ \ldots X_{i+(m-1)J}\right), \tag{C.3}$$

donde J es el retraso o el retardo de reconstrucción, y m es la dimensión embebida. Así, $\mathbf{X}$ es una matriz $M \times m$, y las constantes m, M, J, y N están relacionadas de la siguiente manera:

$$M = N - (m-1)J. \tag{C.4}$$

A partir de la definición de λ_1 dada en la ecuación (C.1), se asume que el j-ésimo par de vecinos cercanos diverge aproximadamente a una taza dada por el máximo exponente de Lyapunov:

$$d_j(i) \approx C_j e^{\lambda_1(i\triangle t)}, \tag{C.5}$$

donde C_j es la separación inicial. Tomando el logaritmo en ambos lados de la ecuación (C.5), se obtiene:

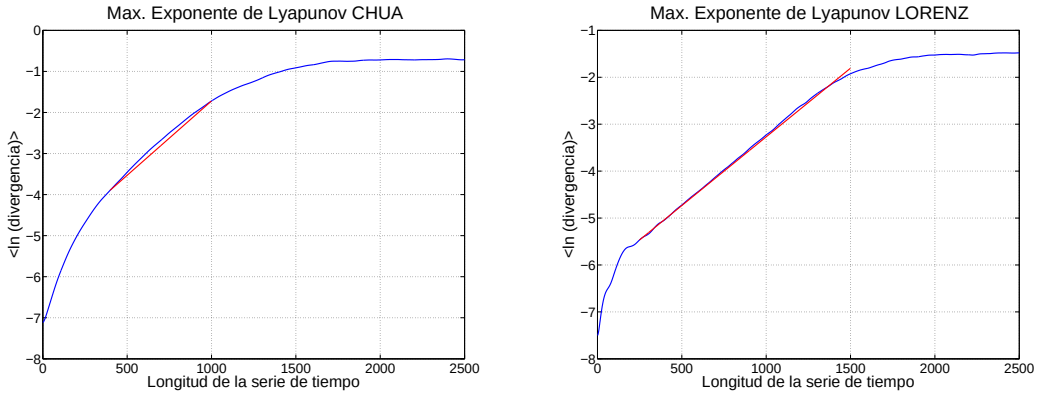
$$\ln d_j(i) \approx \ln C_j + \lambda_1(i \triangle t). \tag{C.6}$$

La ecuación (C.6) representa un conjunto de líneas paralelas aproximadas (para $j = 1, 2, ..., M$), cada uno con una pendiente más o menos proporcional a λ_1. El máximo exponente de Lyapunov es fácilmente calculado usando un ajuste por mínimos cuadrados a la línea media definida por:

$$y(i) = \frac{1}{\triangle t}\langle \ln d_j(i) \rangle, \tag{C.7}$$

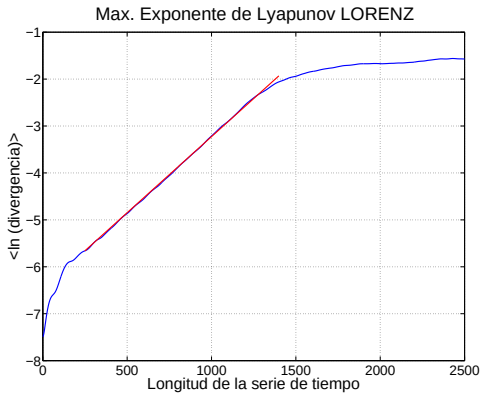
donde $\langle\ \rangle$ denota el promedio de todos los valores de j. Este proceso de promediado es la llave para calcular con exactitud los valores de λ_1 utilizando un pequeño conjunto de datos. Nótese que en la ecuación (C.7), C_j realiza la función de normalización de la separación de los vecinos, pero como se muestra en la ecuación (C.6), esta normalización es innecesaria para estimar λ_1 [34].

De esta manera, con ayuda de la herramienta Tisean 3.0.0 y MATLAB se realiza una aproximación a la pendiente proporcional a λ_1 mediante series de tiempo, obteniendo así los exponentes de Lyapunov de los sistemas caóticos que se utilizaron.

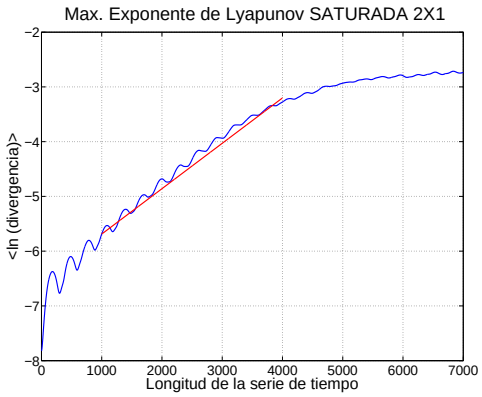


(a) Chua. (b) Lorenz.

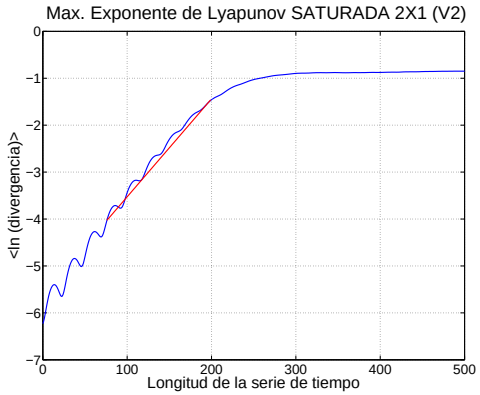
Figura C.1: Aproximación del máximo exponente de Lyapunov mediante series de tiempo.



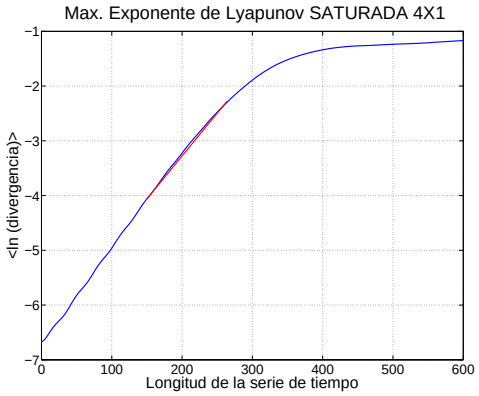
(a) Lorenz $\gamma = 28$.



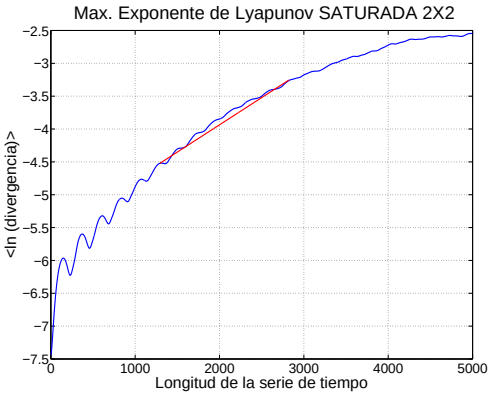
(b) Función saturada 2x1.



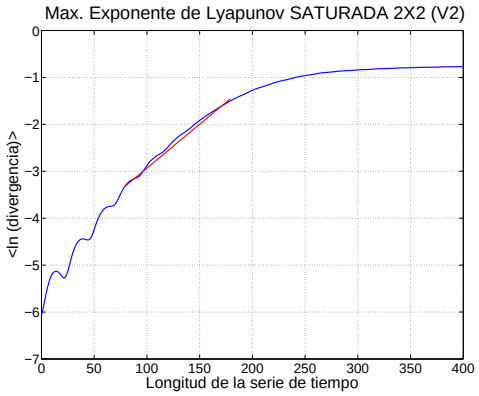
(c) Función saturada 2x1 (V2).



(d) Función saturada 4x1.

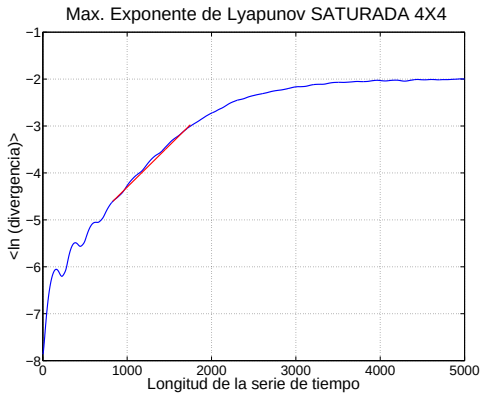


(e) Función saturada 2x2.

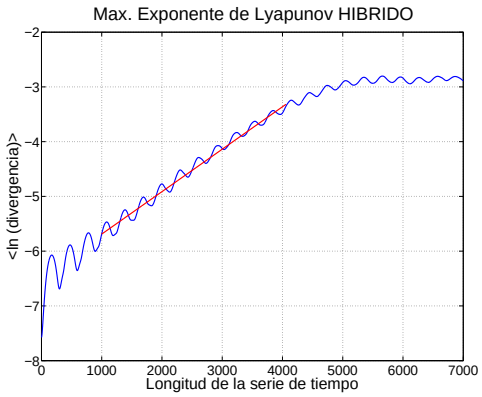


(f) Función saturada 2x2 (V2).

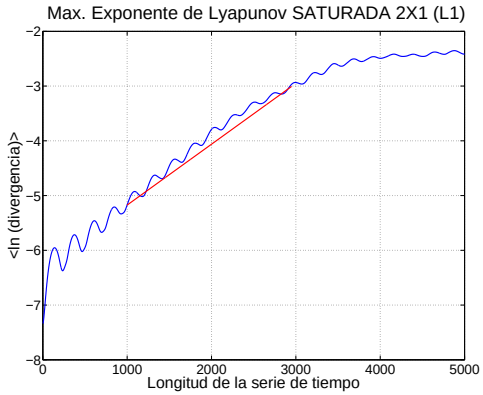
Figura C.2: Aproximación del máximo exponente de Lyapunov mediante series de tiempo.



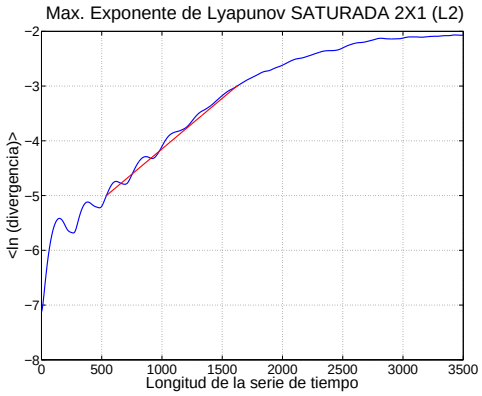
(a) Función saturada 4x4.



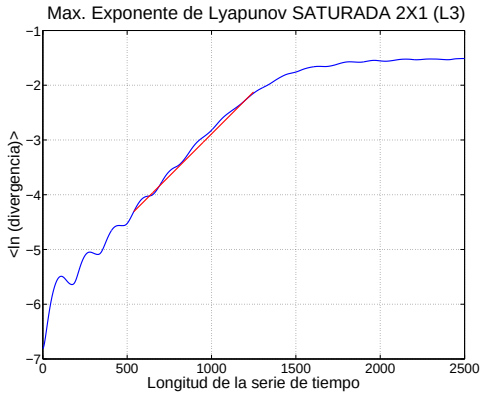
(b) Sistema Híbrido.



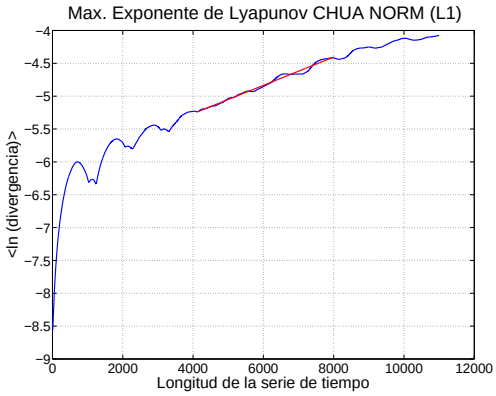
(c) Función saturada 2x1 (L1).



(d) Función saturada 2x1 (L2).

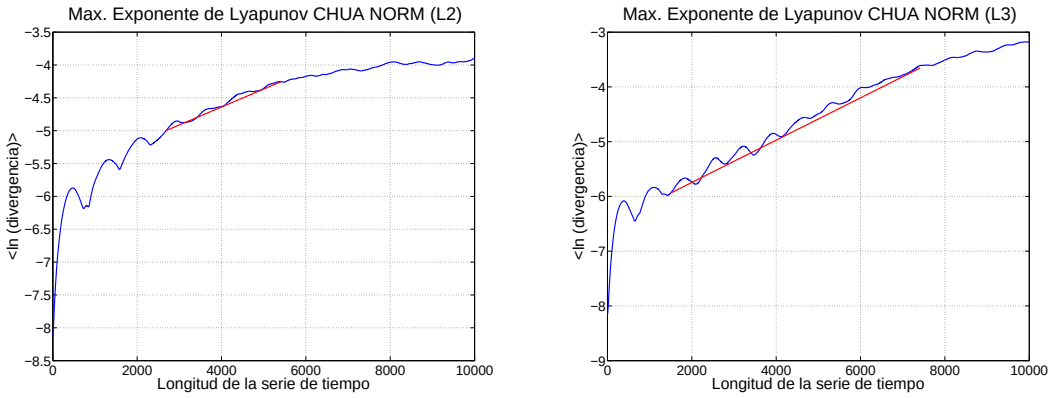


(e) Función saturada 2x1 (L3).



(f) Chua normalizado (L1).

Figura C.3: Aproximación del máximo exponente de Lyapunov mediante series de tiempo.



(a) Chua normalizado (L2). (b) Chua normalizado (L3).

Figura C.4: Aproximación del máximo exponente de Lyapunov mediante series de tiempo.

Sistema	**Max. Exp. de Lyapunov**
Circuito de Chua	0.0036
Sistema de Lorenz	0.0029
Sistema de Lorenz $\gamma = 28$	0.0032
Sistema basado en una función PWL de 2x1	0.00082
Sistema basado en una función PWL de 2x1 (V2)	0.0209
Sistema basado en una función PWL de 4x1	0.0157
Sistema basado en una función PWL de 2x2	0.000829
Sistema basado en una función PWL de 2x2 (V2)	0.0187
Sistema basado en una función PWL de 4x4	0.0018
Sistema Híbrido	0.000752
Chua Normalizado (L1)	0.000215
Chua Normalizado (L2)	0.000273
Chua Normalizado (L3)	0.000386
Saturada 2x1 (L1)	0.0011
Saturada 2x1 (L2)	0.0018
Saturada 2x1 (L3)	0.0031

Cuadro C.1: Cálculo del máximo exponente de Lyapunov mediante series de tiempo.

Apéndice D

Pruebas estadísticas de aleatoriedad

La aleatoriedad es una propiedad probabilística, es decir, las propiedades de una secuencia aleatoria pueden ser caracterizadas y descritas en términos de probabilidad. Hay un número infinito de posibles pruebas estadísticas, cada una evalúa la presencia o ausencia de un "patrón" que, si se detecta, indicaría que la secuencia no es aleatoria. Debido a que hay tantas pruebas para juzgar si una secuencia es aleatoria o no, ningún conjunto finito específico de pruebas se considera "completa". Además, los resultados de las pruebas estadísticas deben interpretarse con precaución para evitar conclusiones erróneas.

Una prueba estadística está formulada para probar una hipótesis nula específica (H0), la cual implica que la secuencia que está siendo probada es *aleatoria*. Asociada a esta hipótesis nula está la hipótesis alternativa (Ha), la cual implica que la secuencia no es aleatoria. Para cada prueba aplicada, una decisión o conclusión se deriva de aceptar o rechazar la hipótesis nula, es decir, si el generador está produciendo valores aleatorios o no.

Si los datos son aleatorios puede ocurrir una conclusión de rechazo a la hipótesis nula (es decir que se concluya que los datos no son aleatorios) en un pequeño porcentaje de los casos, a esta conclusión se le denomina error de tipo I. Si los datos son no aleatorios puede ocurrir una conclusión de aceptación a la hipótesis nula (es decir que se concluya que los datos son aleatorios), y es llamado error de tipo II. La probabilidad de que ocurra un error del tipo I se le llama *nivel de significancia de la prueba* y se denota como α. Para la prueba, α es la probabilidad de que la prueba indique que la secuencia no es aleatoria cuando realmente lo es. Los valores comúnes de α en criptografía son alrededor de 0.01.

La probabilidad de un error de tipo II se denota como β. Para la prueba, β es la probabilidad de que la prueba indique que la secuencia es aleatoria cuando realmente no lo es. A diferencia de α, β no es un valor fijo, β puede asumir muchos valores diferentes, esto debido a que hay un número infinito de formas en que una cadena de datos puede ser no aleatoria, y en cada forma diferente se obtiene un valor de β diferente. El cálculo de β del error de tipo II es más difícil que el cálculo de α debido a que existen muchos tipos posibles de no aleatoriedad.

La salida de las pruebas estadísticas se denomina *P-value*, donde p es un número entre [0,1] y está relacionado con el nivel de significancia α [32]. Entonces un $p - value \geq 0.01$ significaría que la secuencia se consideraría aleatoria con una confianza de 99 %, mientras que un $p - value < 0.01$ significaría que la secuencia no es aleatoria con una confianza del 99 % [20].

De esta manera una prueba es interpretada de la siguiente manera:

$$\begin{aligned} p - value < \alpha &\longrightarrow \textit{prueba no pasada}, \\ p - value \geq \alpha &\longrightarrow \textit{prueba pasada}. \end{aligned}$$

D.1. Paquete de pruebas estadísticas NIST

El NIST es un paquete estadístico que consta de 15 pruebas que se desarrollaron para probar la aleatoriedad de secuencias binarias (de longitud arbitraria) producidas por generadores de números aleatorios o pseudoaleatorios criptográficos basados en hardware o software [20]. Estas pruebas se centran en una variedad de diferentes tipos de no-aleatoriedad que pudieran existir en una secuencia. Las 15 pruebas son:

1. Prueba de frecuencia (monobit).
2. Prueba de frecuencia dentro de un bloque.
3. Pruebas de tramas (Runs Test).
4. Prueba para la trama más larga de unos en un bloque.
5. Prueba de la matriz binaria de rango.
6. Prueba de la transformada discreta de Fourier (Espectro).
7. Prueba comparación de plantillas de no-superposición.
8. Prueba de comparación de plantillas de superposición.
9. Prueba Maurer "Estadística universal".
10. Prueba de complejidad lineal.
11. Prueba serial.
12. Prueba de Entropía aproximada.
13. Prueba de sumas acumulativas (Cusums).
14. Prueba de excursiones aleatorias.
15. Prueba de excursiones aleatorias variantes.

El orden de la aplicación de las pruebas es arbitrario. Sin embargo, se recomienda que la prueba de frecuencia se ejecute primero, ya que esta proporciona la evidencia básica para la existencia de no aleatoriedad en una secuencia, concretamente, la no uniformidad. Si esta prueba falla, la probabilidad de que otras pruebas fallen es alta.

D.1.1. Prueba de la frecuencia (monobit)

El enfoque de esta prueba es la proporción de ceros y unos de toda la secuencia. Determina si el número de unos y ceros en la secuencia es aproximadamente el mismo como sería de esperar para una secuencia verdaderamente aleatoria. La prueba evalúa la cercanía de la fracción de unos a 1/2, es decir, el número de unos y ceros en una secuencia debe ser aproximadamente la misma. Todas las pruebas posteriores dependen de la aprobación de esta prueba. Para esta prueba se recomienda que cada secuencia a ser evaluada conste de un mínimo de 100 bits ($n \geq 100$).

D.1.2. Prueba de frecuencia dentro de un bloque

El enfoque de la prueba es la proporción de unos dentro de los bloques de M-bits. Determina si la frecuencia de unos en un bloque de M bits es de aproximadamente $M/2$, como sería de esperar bajo un supuesto de aleatoriedad. Con el tamaño del bloque $M = 1$ la prueba es equivalente a la prueba de la frecuencia (monobit). Para esta prueba, al igual que la del monobit, se recomienda que cada secuencia que va a ser evaluada conste de un mínimo de 100 bits ($n \geq 100$).

D.1.3. Prueba de tramas

El enfoque de esta prueba es el número total de tramas en la secuencia, donde una trama es una secuencia ininterrumpida de bits idénticos. Una trama de longitud k consiste en exactamente k bits idénticos y está limitada antes y después con un bit de valor opuesto. El propósito de la prueba de tramas es para determinar si el número de tramas de unos y ceros de diferentes longitudes es el esperado para una secuencia aleatoria. En particular, esta prueba determina si la oscilación entre estos ceros y unos es demasiado rápido o demasiado lento. Para esta prueba, al igual que las pruebas anteriores, se recomienda que cada secuencia que va a ser evaluada conste de un mínimo de 100 bits ($n \geq 100$).

D.1.4. Prueba para la trama más larga de unos en un bloque

El objetivo de la prueba es obtener la trama más larga de unos dentro de los bloques de M-bits. El propósito es determinar si la longitud de la trama más larga de unos dentro de la secuencia es coherente con la longitud de la trama más larga de unos que se esperaría en una secuencia aleatoria. Tenga en cuenta que una irregularidad en la longitud prevista de la trama más larga de unos implica que también hay una irregularidad en la longitud prevista de la trama más larga de ceros. Por lo tanto, sólo una prueba para unos es necesario.

El código de la prueba ha sido preestablecido para dar cabida a tres valores para la longitud de cada bloque M: $M = 8$, $M = 128$ y $M = 10^4$, de acuerdo con los siguientes valores de la longitud de la secuencia n:

Mínimo n	M
128	8
6272	128
750000	10^4

Cuadro D.1: Valores de M de acuerdo a n.

Se recomienda que cada secuencia conste de un mínimo de bits como se especifica en la Tabla D.1.

D.1.5. Prueba de la matriz binaria de rango

El enfoque de la prueba consta en el rango de sub-matrices disjuntas de toda la secuencia. El propósito es comprobar si hay dependencia lineal entre subseries de longitud fija de la secuencia original. Las probabilidades para $M = Q = 32$ se han calculado y se insertan en el código de prueba. Se pueden seleccionar otros valores de M y Q, pero tendrían que ser recalculadas las probabilidades. El número mínimo de bits debe ser tal que $n \geq 38MQ$ (es decir, se crean al menos 38 matrices). Para $M = Q = 32$, cada secuencia a ser probada debe constar de un mínimo de $38,912$ bits donde:

n es la longitud de la cadena de bits.
M corresponde al número de filas de cada matriz.
Q es el número de columnas en cada matriz.

D.1.6. Prueba de la Transformada Discreta de Fourier (espectro)

Esta prueba toma las alturas de los picos de la Transformada Discreta de Fourier de la secuencia. El propósito de esta prueba es detectar características periódicas en la secuencia de prueba que indique una desviación de la hipótesis de aleatoriedad. La intención es detectar si el número de picos que superan el umbral del 95 % es significativamente diferente de 5 %. Se recomienda que cada secuencia a ser probada tenga como mínimo 1000 bits ($n \geq 1000$).

D.1.7. Prueba comparación de plantillas de no superposición

El propósito de esta prueba es detectar los generadores que producen demasiadas ocurrencias de un patrón no periódico dado. Para esta prueba y para la prueba de plantilla de superposición, se utiliza una ventana de m bits para buscar un patrón específico de m bits. Si no se encuentra el patrón, la ventana se desliza una posición de un bit. Si se encuentra el patrón, la ventana se resetea y la búsqueda se reanuda. Cada secuencia a ser analizada debe contener un mínimo de 10^6 bits ($n > 10^6$).

D.1.8. Prueba de comparación de plantillas de superposición

El enfoque de la prueba de comparación de plantillas de superposición es el número de apariciones de las cadenas de destino especificados previamente. Al igual que la prueba anterior, utiliza una ventana de m bits para buscar un patrón específico de m bits, si no se encuentra el patrón, la ventana se desliza una posición de un bit. La diferencia entre esta prueba y la prueba anterior es que cuando se encuentra el patrón, la ventana se desliza sólo un bit antes de reanudar la búsqueda. Cada secuencia a ser analizada debe contener un mínimo de 10^6 bits ($n > 10^6$).

D.1.9. Prueba de Maurer Estadística Universal

Esta prueba se enfoca en el número de bits entre patrones que coinciden (una medida que está relacionada con la longitud de una secuencia comprimida). El propósito de la prueba es detectar si la secuencia se puede o no comprimir de manera significativa sin pérdida de información. Una secuencia significativamente compresible se considera que es no aleatoria. Esta prueba requiere una larga secuencia de bits ($n \geq (Q + K)L$), donde:

L: La longitud de cada bloque.
Q: El número de bloques en la secuencia de inicialización.
n: La longitud de la cadena de bits.

Mínimo n	L	$Q = 10 \cdot 2^L$
387840	6	640
904960	7	1280
2068480	8	2560
4654080	9	5120
10342400	10	10240
22753280	11	20480
49643520	12	40960
107560960	13	81920
231669760	14	163840
496435200	15	327680
1059061760	16	655360

Cuadro D.2: Valores de L, Q y n.

Los valores de L, Q y n deben ser elegidos de acuerdo a la Tabla D.2.

D.1.10. Prueba de complejidad lineal

Esta prueba se enfoca en analizar la longitud de un registro de desplazamiento de retroalimentación lineal (LFSR). El propósito es determinar si la secuencia es o no lo suficientemente

compleja como para ser considerado aleatorio. Las secuencias aleatorias se caracterizan por LFSRs más largos. Un LFSR que es demasiado corto implica no aleatoriedad. La prueba requiere una secuencia de bits $n \geq 10^6$.

D.1.11. Prueba serial

El enfoque de esta prueba es analizar la frecuencia de todos los posibles patrones de superposición de m bits a través de toda la secuencia. El propósito es determinar si el número de ocurrencias de los 2^m patrones m-bits superpuestos es aproximadamente el mismo que el que se esperaría para una secuencia aleatoria. Las secuencias aleatorias tienen uniformidad; es decir, cada patrón de m-bits tiene la misma oportunidad de aparecer como cualquier otro patrón de m bits. Se debe de tener en cuenta que para $m = 1$, la prueba serial es equivalente a la prueba del monobit. Por lo tanto es necesario elegir m y n tales que $m < [\log_2 n] - 2$.

D.1.12. Prueba de Entropía aproximada

Al igual que con la prueba serial, el objetivo de esta prueba es determinar la frecuencia de todos los posibles patrones de superposición de m bits a través de toda la secuencia. El propósito de la prueba es comparar la frecuencia de superposición de bloques de dos longitudes consecutivas/adyacentes (m y $m + 1$) contra el resultado esperado por la superposición de una secuencia aleatoria. Es necesario elegir m y n tales que $m < [\log_2 n] - 5$.

D.1.13. Prueba de sumas acumulativas (Cusums)

El enfoque de esta prueba es analizar la excursión máxima (de cero) del camino aleatorio definido por la suma acumulada de dígitos ajustados $(-1, 1)$ de la secuencia. El propósito de la prueba es determinar si la suma acumulada de las secuencias parciales que se producen en la secuencia de prueba es demasiado grande o demasiado pequeño en relación con el comportamiento esperado de esa suma acumulativa para secuencias aleatorias. Esta suma acumulativa puede ser considerada como un camino aleatorio. Para una secuencia aleatoria, las excursiones deben estar cerca de cero. Para ciertos tipos de secuencias no aleatorias, las excursiones desde cero serán grandes. La aplicación de la prueba se realiza hacia adelante a través de la secuencia de entrada ($modo = 0$) o hacia atrás a través de la secuencia ($modo = 1$). Se recomienda que cada secuencia a ser probada conste de un mínimo de 100 bits ($n \geq 100$).

D.1.14. Prueba de excursiones aleatorias

El objetivo de esta prueba consiste en determinar el número de ciclos que tienen exactamente K visitas en una suma acumulada. La suma acumulada se deriva de sumas parciales después de que la secuencia (0,1) se transfiere a la secuencia apropiada $(-1, 1)$. Un ciclo consiste en una secuencia de pasos de unidad de longitud tomadas al azar que comienzan y regresan al origen. El propósito es determinar si el número de visitas a un estado en particular dentro de un ciclo se desvía de lo que uno esperaría de una secuencia aleatoria. Esta prueba es en realidad una serie de ocho pruebas (y conclusiones), una prueba y conclusión para cada uno de los estados:

$-4, -3, -2, -1$ y $1, 2, 3, 4$. Se recomienda que cada secuencia a ser probada conste de un mínimo de 1,000,000 bits ($n \geq 10^6$).

D.1.15. Prueba de excursiones aleatorias variantes

El objetivo de esta prueba es determinar el número total de veces que se visita un estado en particular en una suma acumulada. El propósito es detectar desviaciones del número de visitas esperadas. Esta prueba en realidad consta de 18 pruebas (y conclusiones), un prueba y una conclusión de cada uno de los estados: $-9, -8, ..., -1$ y $1, 2, ..., 9$. Se recomienda que cada secuencia a ser probada conste de un mínimo de 1,000,000 bits ($n \geq 10^6$).

D.2. Pruebas NIST de los RNGs diseñados

En esta sección se realizan las pruebas de aleatoriedad del paquete NIST, para medir estadísticamente la distribución de los bits generados. Como se mencionó anteriormente algunas pruebas necesitan cadenas muy largas de bits ($n \geq 10^6$). Sin embargo debido a la cantidad de RNGs diseñados y al número de iteraciones necesarias para la obtención de un millón de bits no fue posible obtener esa cadena en todos los RNGs. Así que la prueba en un millón de bits solo se llevó a cabo en algunos de ellos. A continuación se presentan las tablas de las pruebas estadísticas y el *p-value* de cada una de ellas.

RNG CHUA	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.703945	0.282297	0.667196
Frecuencia (bloque)	0.998263	1.000000	1.000000
Runs	0.000000	0.000000	0.000000
Longest Run	0.000000	0.000000	0.000000
Matríz binaria de rango	0.374306	0.663163	0.904959
Transformada discreta de Fourier	0.000000	0.000000	0.000000
Plantillas de no-superposición	SUCCESS	FAILURE	FAILURE
Plantillas superposición	0.008551	0.000000	0.000000
Maurer	-	-	0.000000
Complejidad lineal	0.053594	0.106543	0.957247
Prueba serial	0.000000 0.093628	0.000000 0.000000	0.000000 0.000000
Entropía aproximada	0.000000	0.000000	0.000000
Sumas acumulativas	0.675485 0.971436	0.543022 0.511215	0.830661 0.985512
Excursiones aleatorias	0.000000	0.000000	0.000000
Excursiones aleatorias variantes	0.000000	0.000000	0.000000

Cuadro D.3: Resultados del NIST obtenidos del RGN basado en el sistema de Chua.

RNG CHUA (VN)	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.952156	0.879353	0.090646
Frecuencia (bloque)	0.987399	1.000000	1.000000
Runs	0.000000	0.000000	0.000000
Longest Run	0.000001	0.000000	0.000000
Matríz binaria de rango	0.648387	0.204856	0.397831
Transformada discreta de Fourier	0.027638	0.353091	0.000000
Plantillas de no-superposición	SUCCESS	SUCCESS	FAILURE
Plantillas superposición	0.008551	0.000000	0.000000
Maurer	-	-	0.000000
Complejidad lineal	0.676533	0.430138	0.142563
Prueba serial	0.083785 0.071858	0.001657 0.143680	0.000000 0.574834
Entropía aproximada	0.000027	0.000000	0.000000
Sumas acumulativas	0.998389 0.999845	0.858413 0.721212	0.113041 0.107462
Excursiones aleatorias	0.000000	SUCCESS	SUCCESS
Excursiones aleatorias variantes	0.000000	SUCCESS	SUCCESS

Cuadro D.4: Resultados del NIST obtenidos del RGN basado en el sistema de Chua (VN).

RNG Lorenz ($\gamma = 24$))	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.262714	0.192623	0.369186
Frecuencia (bloque)	0.794498	0.999992	1.000000
Runs	0.000000	0.000000	0.000000
Longest Run	0.000000	0.000000	0.000000
Matríz binaria de rango	0.157494	0.509536	0.657952
Transformada discreta de Fourier	0.098577	0.383988	0.002775
Plantillas de no-superposición	SUCCESS	SUCCESS	FAILURE
Plantillas superposición	0.008551	0.000000	0.000000
Maurer	-	-	0.000000
Complejidad lineal	0.856990	0.799895	0.809173
Prueba serial	0.560035 0.844668	0.005653 0.349415	0.000000 0.126272
Entropía aproximada	0.000738	0.000000	0.000000
Sumas acumulativas	0.436946 0.178261	0.345715 0.274399	0.696084 0.311757
Excursiones aleatorias	FAILURE	FAILURE	FAILURE
Excursiones aleatorias variantes	FAILURE	FAILURE	FAILURE

Cuadro D.5: Resultados del NIST obtenidos del RGN basado en el sistema de Lorenz.

RNG Lorenz ($\gamma = 28$))	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.522173	0.785656	0.607252
Frecuencia (bloque)	0.998161	1.000000	1.000000
Runs	0.000000	0.000000	0.000000
Longest Run	0.000000	0.000000	0.000000
Matríz binaria de rango	0.004936	0.707940	0.798172
Transformada discreta de Fourier	1.000000	0.045251	0.000000
Plantillas de no-superposición	SUCCESS	SUCCESS	FAILURE
Plantillas superposición	0.008551	0.000000	0.000000
Maurer	-	-	0.000000
Complejidad lineal	0.133284	0.753494	0.181140
Prueba serial	0.498961 0.711796	0.000109 0.356793	0.000000 0.626960
Entropía aproximada	0.000477	0.000000	0.000000
Sumas acumulativas	0.936213 0.483106	0.345715 0.709326	0.898063 0.884469
Excursiones aleatorias	0.000000	SUCCESS	SUCCESS
Excursiones aleatorias variantes	0.000000	SUCCESS	SUCCESS

Cuadro D.6: Resultados del NIST obtenidos del RGN basado en el sistema de Lorenz.

RNG PWL 2x1	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.561915	0.440354	0.674485
Frecuencia (bloque)	0.869021	1.000000	1.000000
Runs	0.000000	0.000000	0.000000
Longest Run	0.000030	0.000000	0.000000
Matríz binaria de rango	0.159044	0.502805	0.605379
Transformada discreta de Fourier	0.462869	0.642429	0.000000
Plantillas de no-superposición	SUCCESS	SUCCESS	FAILURE
Plantillas superposición	0.008551	0.000000	0.000000
Maurer	-	-	0.000000
Complejidad lineal	0.959424	0.547903	0.124244
Prueba serial	0.954913 0.997815	0.008160 0.297005	0.000000 0.932363
Entropía aproximada	0.000004	0.0000000	0.000000
Sumas acumulativas	0.620100 0.874256	0.820335 0.272731	0.944235 0.853792
Excursiones aleatorias	0.000000	0.000000	SUCCESS
Excursiones aleatorias variantes	0.000000	0.000000	SUCCESS

Cuadro D.7: Resultados del NIST obtenidos del RGN basado en un sistema PWL 2x1.

RNG PWL 2x1 (V2)	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.447255	0.785656	0.332046
Frecuencia (bloque)	0.900732	1.000000	1.000000
Runs	0.000000	0.000000	0.000000
Longest Run	0.000002	0.000000	0.000000
Matríz binaria de rango	0.374306	0.199620	0.033510
Transformada discreta de Fourier	0.021781	0.045251	0.000000
Plantillas de no-superposición	SUCCESS	SUCCESS	FAILURE
Plantillas superposición	0.008551	0.000000	0.000000
Maurer	-	-	0.000000
Complejidad lineal	0.423067	0.444963	0.365196
Prueba serial	0.378448 0.116398	0.000632 0.697999	0.000000 0.303528
Entropía aproximada	0.000759	0.0000000	0.000000
Sumas acumulativas	0.656867 0.387010	0.891084 0.647327	0.637475 0.106226
Excursiones aleatorias	0.000000	0.000000	SUCCESS
Excursiones aleatorias variantes	0.000000	0.000000	SUCCESS

Cuadro D.8: Resultados del NIST obtenidos del RGN basado en el sistema PWL 2x1.

RNG PWL 4x1	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.841481	0.088886	0.190196
Frecuencia (bloque)	0.819497	1.000000	1.000000
Runs	0.057380	0.000009	0.000000
Longest Run	0.000023	0.000000	0.000000
Matríz binaria de rango	0.949536	0.199951	0.238095
Transformada discreta de Fourier	0.021781	0.705990	0.000048
Plantillas de no-superposición	SUCCESS	SUCCESS	FAILURE
Plantillas superposición	0.008551	0.000000	0.000000
Maurer	-	-	0.000000
Complejidad lineal	0.000003	0.605253	0.935749
Prueba serial	0.712745 0.812251	0.000033 0.044223	0.000000 0.075816
Entropía aproximada	0.012847	0.0000000	0.000000
Sumas acumulativas	0.931330 0.975089	0.056382 0.139002	0.292977 0.356497
Excursiones aleatorias	0.000000	0.000000	0.000000
Excursiones aleatorias variantes	0.000000	0.000000	0.000000

Cuadro D.9: Resultados del NIST obtenidos del RGN basado en el sistema PWL 4x1.

RNG PWL 2x2 (V1)	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.689157	0.125883	0.000025
Frecuencia (bloque)	0.894525	1.000000	1.000000
Runs	0.000000	0.000000	0.000000
Longest Run	0.000000	0.000000	0.000000
Matríz binaria de rango	0.037210	0.509536	0.293617
Transformada discreta de Fourier	0.581909	0.009803	0.000000
Plantillas de no-superposición	SUCCESS	FAILURE	FAILURE
Plantillas superposición	0.008551	0.000000	0.000000
Maurer	-	-	0.000000
Complejidad lineal	0.868349	0.232938	0.517092
Prueba serial	0.814738 0.527403	0.000000 0.191221	0.000000 0.069580
Entropía aproximada	0.000000	0.0000000	0.000000
Sumas acumulativas	0.849583 0.943118	0.232058 0.217745	0.000035 0.000046
Excursiones aleatorias	0.000000	0.000000	0.000000
Excursiones aleatorias variantes	0.000000	0.000000	0.000000

Cuadro D.10: Resultados del NIST obtenidos del RGN basado en el sistema PWL 2x1.

RNG PWL 2x2 (V2)	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.447255	0.785656	0.332046
Frecuencia (bloque)	0.900732	1.000000	1.000000
Runs	0.000000	0.000000	0.000000
Longest Run	0.000002	0.000000	0.000000
Matríz binaria de rango	0.374306	0.199620	0.033510
Transformada discreta de Fourier	0.021781	0.045251	0.000000
Plantillas de no-superposición	SUCCESS	SUCCESS	FAILURE
Plantillas superposición	0.008551	0.000000	0.000000
Maurer	-	-	0.000000
Complejidad lineal	0.423067	0.444963	0.365196
Prueba serial	0.378448 0.116398	0.000632 0.697999	0.000000 0.303528
Entropía aproximada	0.000000	0.0000000	0.000000
Sumas acumulativas	0.656867 0.387010	0.891084 0.647327	0.637475 0.106226
Excursiones aleatorias	0.000000	0.000000	SUCCESS
Excursiones aleatorias variantes	0.000000	0.000000	SUCCESS

Cuadro D.11: Resultados del NIST obtenidos del RGN basado en el sistema PWL 2x2.

RNG PWL 4x4	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.936237	0.393208	0.219447
Frecuencia (bloque)	0.899204	1.000000	1.000000
Runs	0.000000	0.000000	0.000000
Longest Run	0.000000	0.000000	0.000000
Matríz binaria de rango	0.862457	0.429607	0.486254
Transformada discreta de Fourier	0.027638	0.034142	0.000000
Plantillas de no-superposición	SUCCESS	SUCCESS	FAILURE
Plantillas superposición	0.008551	0.000000	0.000000
Maurer	-	-	0.000000
Complejidad lineal	0.157341	0.248492	0.366178
Prueba serial	0.178264 0.426779	0.000000 0.320500	0.000000 0.990061
Entropía aproximada	0.000017	0.0000000	0.000000
Sumas acumulativas	0.874256 0.805785	0.618347 0.372302	0.202009 0.337570
Excursiones aleatorias	0.000000	0.000000	0.000000
Excursiones aleatorias variantes	0.000000	0.000000	0.000000

Cuadro D.12: Resultados del NIST obtenidos del RGN basado en el sistema PWL 4x4.

RNG Chua-Lorenz	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.535258	0.785656	0.793321
Frecuencia (bloque)	0.936282	1.000000	1.000000
Runs	0.132607	0.000000	0.000000
Longest Run	0.002515	0.000000	0.000000
Matríz binaria de rango	0.587007	0.310099	0.445531
Transformada discreta de Fourier	0.646355	0.001724	0.003419
Plantillas de no-superposición	SUCCESS	SUCCESS	FAILURE
Plantillas superposición	0.008551	0.000000	0.000000
Maurer	-	-	0.000000
Complejidad lineal	0.910170	0.171255	0.681848
Prueba serial	0.026285 0.151131	0.000000 0.010245	0.000000 0.454524
Entropía aproximada	0.000000	0.0000000	0.000000
Sumas acumulativas	0.874256 0.475200	0.847797 0.992505	0.954761 0.993709
Excursiones aleatorias	0.000000	SUCCESS	SUCCESS
Excursiones aleatorias variantes	0.000000	SUCCESS	SUCCESS

Cuadro D.13: Resultados del NIST obtenidos del RGN Chua-Lorenz.

RNG Chua-Saturada	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.289145	0.844562	0.824314
Frecuencia (bloque)	0.998282	1.000000	1.000000
Runs	0.000000	0.000000	0.000000
Longest Run	0.000161	0.000000	0.000000
Matríz binaria de rango	0.374306	0.393345	0.300675
Transformada discreta de Fourier	0.000000	0.000000	0.000000
Plantillas de no-superposición	SUCCESS	FAILURE	FAILURE
Plantillas superposición	0.008551	0.000000	0.000000
Maurer	-	-	0.000000
Complejidad lineal	0.543677	0.883706	0.190283
Prueba serial	0.000000 0.0000001	0.000000 0.000000	0.000000 0.000000
Entropía aproximada	0.000000	0.0000000	0.000000
Sumas acumulativas	0.373520 0.566584	0.946993 0.797610	0.831535 0.626481
Excursiones aleatorias	0.000000	0.000000	FAILURE
Excursiones aleatorias variantes	0.000000	0.000000	SUCCESS

Cuadro D.14: Resultados del NIST obtenidos del RGN Chua-Saturada.

RNG Lorenz-Chua	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.123560	0.054522	-
Frecuencia (bloque)	0.952130	1.000000	-
Runs	0.026132	0.000000	-
Longest Run	0.000073	0.000000	-
Matríz binaria de rango	0.587007	0.973521	-
Transformada discreta de Fourier	0.581909	0.201659	-
Plantillas de no-superposición	SUCCESS	SUCCESS	-
Plantillas superposición	0.008551	0.000000	-
Maurer	-	-	-
Complejidad lineal	0.484284	0.651158	-
Prueba serial	0.427901 0.398638	0.010710 0.966217	- -
Entropía aproximada	0.001504	0.0000000	-
Sumas acumulativas	0.197884 0.131536	0.051155 0.087280	- -
Excursiones aleatorias	0.000000	0.000000	-
Excursiones aleatorias variantes	0.000000	0.000000	-

Cuadro D.15: Resultados del NIST obtenidos del RGN Lorenz-Chua.

RNG Lorenz-Saturada	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.207669	0.506640	-
Frecuencia (bloque)	0.966028	1.000000	-
Runs	0.000000	0.000000	-
Longest Run	0.000000	0.000000	-
Matríz binaria de rango	0.159044	0.821438	-
Transformada discreta de Fourier	0.520637	0.000554	-
Plantillas de no-superposición	SUCCESS	SUCCESS	-
Plantillas superposición	0.008551	0.000000	-
Maurer	-	-	-
Complejidad lineal	0.649483	0.688769	-
Prueba serial	0.000078 0.022841	0.000000 0.591671	- -
Entropía aproximada	0.000000	0.0000000	-
Sumas acumulativas	0.322973 0.237514	0.411511 0.850473	- -
Excursiones aleatorias	0.000000	0.000000	-
Excursiones aleatorias variantes	0.000000	0.000000	-

Cuadro D.16: Resultados del NIST obtenidos del RGN Lorenz-Saturada.

RNG Saturada-Chua	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.389789	0.849515	-
Frecuencia (bloque)	0.964845	0.999902	-
Runs	0.553417	0.000554	-
Longest Run	0.004550	0.000000	-
Matríz binaria de rango	0.014014	0.053736	-
Transformada discreta de Fourier	0.142033	0.727669	-
Plantillas de no-superposición	SUCCESS	SUCCESS	-
Plantillas superposición	0.008551	0.000000	-
Maurer	-	-	-
Complejidad lineal	0.216866	0.606906	-
Prueba serial	0.091889 0.087773	0.000154 0.448865	- -
Entropía aproximada	0.000015	0.0000000	-
Sumas acumulativas	0.507320 0.656867	0.886262 0.985527	- -
Excursiones aleatorias	0.000000	0.000000	-
Excursiones aleatorias variantes	0.000000	0.000000	-

Cuadro D.17: Resultados del NIST obtenidos del RGN Saturada-Chua.

RNG Saturada-Lorenz	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.825871	0.410968	-
Frecuencia (bloque)	0.987186	1.000000	-
Runs	0.904866	0.000833	-
Longest Run	0.000034	0.000000	-
Matríz binaria de rango	0.022669	0.521160	-
Transformada discreta de Fourier	0.581909	0.055460	-
Plantillas de no-superposición	SUCCESS	SUCCESS	-
Plantillas superposición	0.008551	0.000000	-
Maurer	-	-	-
Complejidad lineal	0.879372	0.951261	-
Prueba serial	0.721432 0.853213	0.000525 0.268962	- -
Entropía aproximada	0.000083	0.0000000	-
Sumas acumulativas	0.897326 0.981486	0.615476 0.436852	- -
Excursiones aleatorias	0.000000	0.000000	-
Excursiones aleatorias variantes	0.000000	0.000000	-

Cuadro D.18: Resultados del NIST obtenidos del RGN Saturada-Lorenz.

RNG Chua normalizado L1	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.435391	0.293775	-
Frecuencia (bloque)	0.643782	1.000000	-
Runs	0.000003	0.000000	-
Longest Run	0.000001	0.000000	-
Matríz binaria de rango	0.374306	0.914976	-
Transformada discreta de Fourier	0.646355	0.245739	-
Plantillas de no-superposición	SUCCESS	SUCCESS	-
Plantillas superposición	0.008551	0.000000	-
Maurer	-	-	-
Complejidad lineal	0.049827	0.397046	-
Prueba serial	0.658155 0.612659	0.000020 0.271944	- -
Entropía aproximada	0.001525	0.0000000	-
Sumas acumulativas	0.373520 0.769148	0.409259 0.564830	- -
Excursiones aleatorias	0.000000	0.000000	-
Excursiones aleatorias variantes	0.000000	0.000000	-

Cuadro D.19: Resultados del NIST obtenidos del RGN híbrido.

RNG Chua normalizado L1	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.779478	0.964688	0.465390
Frecuencia (bloque)	0.656406	0.999997	1.000000
Runs	0.000075	0.000000	0.000000
Longest Run	0.005701	0.000000	0.000000
Matríz binaria de rango	0.499514	0.532069	0.386459
Transformada discreta de Fourier	0.581909	0.172601	0.005583
Plantillas de no-superposición	SUCCESS	SUCCESS	FAILURE
Plantillas superposición	0.008551	0.000000	0.000000
Maurer	-	-	0.000000
Complejidad lineal	0.757083	0.230983	0.896030
Prueba serial	0.165289 0.074344	0.137839 0.794075	0.000000 0.676384
Entropía aproximada	0.006850	0.0000000	0.000000
Sumas acumulativas	0.507320 0.759852	0.971736 0.985527	0.359726 0.642075
Excursiones aleatorias	0.000000	0.000000	SUCCESS
Excursiones aleatorias variantes	0.000000	0.000000	SUCCESS

Cuadro D.20: Resultados del NIST obtenidos del RGN Chua normalizado L1.

RNG Chua normalizado L2	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.509254	0.595237	-
Frecuencia (bloque)	0.999738	1.000000	-
Runs	0.000493	0.000000	-
Longest Run	0.000000	0.000000	-
Matríz binaria de rango	0.648387	0.127592	-
Transformada discreta de Fourier	0.713570	0.000101	-
Plantillas de no-superposición	SUCCESS	FAILURE	-
Plantillas superposición	0.008551	0.000000	-
Maurer	-	-	-
Complejidad lineal	0.868345	0.203638	-
Prueba serial	0.340255 0.292651	0.000000 0.440292	- -
Entropía aproximada	0.000055	0.0000000	-
Sumas acumulativas	0.629223 0.857965	0.966707 0.529633	- -
Excursiones aleatorias	0.000000	SUCCESS	-
Excursiones aleatorias variantes	0.000000	SUCCESS	-

Cuadro D.21: Resultados del NIST obtenidos del RGN Chua normalizado L2.

RNG Chua normalizado L3	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.718847	0.130644	0.011281
Frecuencia (bloque)	0.984254	0.999113	1.000000
Runs	0.690108	0.544335	0.000000
Longest Run	0.000026	0.000000	0.000000
Matríz binaria de rango	0.203766	0.607817	0.433315
Transformada discreta de Fourier	0.854380	0.884636	0.102376
Plantillas de no-superposición	SUCCESS	SUCCESS	FAILURE
Plantillas superposición	0.008551	0.000000	0.000000
Maurer	-	-	0.000000
Complejidad lineal	0.783222	0.824869	0.770394
Prueba serial	0.685953 0.817120	0.012470 0.740834	0.000000 0.672000
Entropía aproximada	0.000000	0.0000000	0.000000
Sumas acumulativas	0.849583 0.999430	0.186237 0.258072	0.000000 0.000000
Excursiones aleatorias	0.000000	0.000000	0.000000
Excursiones aleatorias variantes	0.000000	0.000000	0.000000

Cuadro D.22: Resultados del NIST obtenidos del RGN Chua normalizado L3.

RNG PWL 2x1 (L1)	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	1.000000	0.017706	0.790239
Frecuencia (bloque)	0.999574	1.000000	1.000000
Runs	0.167587	0.000000	0.000000
Longest Run	0.000271	0.000000	0.000000
Matríz binaria de rango	0.439868	0.117778	0.213544
Transformada discreta de Fourier	1.000000	0.059263	0.000001
Plantillas de no-superposición	SUCCESS	SUCCESS	SUCCESS
Plantillas superposición	0.008551	0.000000	0.000000
Maurer	-	-	0.000000
Complejidad lineal	0.985572	0.440131	0.342594
Prueba serial	0.928153 0.364227	0.000054 0.014889	0.000000 0.024402
Entropía aproximada	0.000759	0.0000000	0.000000
Sumas acumulativas	0.849583 0.849583	0.029787 0.003925	0.681092 0.451991
Excursiones aleatorias	0.000000	0.000000	0.000000
Excursiones aleatorias variantes	0.000000	0.000000	0.000000

Cuadro D.23: Resultados del NIST obtenidos del RGN basado en el sistema PWL 2x1 L1.

RNG Saturada 2x1 L2	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.368120	0.273890	0.005110
Frecuencia (bloque)	0.995030	1.000000	1.000000
Runs	0.000000	0.000000	0.000000
Longest Run	0.000000	0.000000	0.000000
Matríz binaria de rango	0.949536	0.114321	0.498481
Transformada discreta de Fourier	0.168669	0.009803	0.000000
Plantillas de no-superposición	SUCCESS	SUCCESS	FAILURE
Plantillas superposición	0.008551	0.000000	0.000000
Maurer	-	-	0.000000
Complejidad lineal	0.496494	0.676858	0.462577
Prueba serial	0.667525 0.996921	0.000000 0.366943	0.000000 0.818518
Entropía aproximada	0.000000	0.0000000	0.000000
Sumas acumulativas	0.611031 0.193828	0.513823 0.363968	0.007803 0.008055
Excursiones aleatorias	0.000000	0.000000	0.000000
Excursiones aleatorias variantes	0.000000	0.000000	0.000000

Cuadro D.24: Resultados del NIST obtenidos del RGN Saturada 2x1 L2.

RNG Saturada 2x1 L3	**P-Value** ($n = 1e^4$)	**P-Value** ($n = 1e^5$)	**P-Value** ($n = 1e^6$)
Frecuencia (monobit)	0.952156	0.436616	0.858723
Frecuencia (bloque)	0.994677	1.000000	1.000000
Runs	0.000000	0.000000	0.000000
Longest Run	0.000022	0.000000	0.000000
Matríz binaria de rango	0.949536	0.300632	0.137424
Transformada discreta de Fourier	0.168669	0.007591	0.000000
Plantillas de no-superposición	SUCCESS	SUCCESS	FAILURE
Plantillas superposición	0.008551	0.000000	0.000000
Maurer	-	-	0.000000
Complejidad lineal	0.543677	0.514560	0.168613
Prueba serial	0.340255 0.884218	0.000080 0.040825	0.000000 0.961246
Entropía aproximada	0.000005	0.0000000	0.000000
Sumas acumulativas	0.998832 0.993982	0.817526 0.706355	0.914406 0.765435
Excursiones aleatorias	0.000000	SUCCESS	SUCCESS
Excursiones aleatorias variantes	0.000000	SUCCESS	SUCCESS

Cuadro D.25: Resultados del NIST obtenidos del RGN Saturada 2x1 L3.

Apéndice E

Ponencias en congresos y estancia

En este apéndice se muestran las evidencias de la participación en tres congresos: uno nacional y dos internacionales. Se participó en el *XXIX SOMI CONGRESO DE INSTRUMENTACIÓN*, realizado en Puerto Vallarta, Jalisco, México, del 29 al 31 de Octubre de 2014, en la modalidad de ponencia del artículo titulado "Análisis de la simetría de un oscilador caótico basado en un algoritmo numérico de orden adaptivo", cuyo artículo arbitrado fue incluido en las memorias del evento. Además, se presentó el trabajo "Fault conditions of a simple chaotic circuit under capacitor nonlinear effects" en el congreso internacional *LATS 2015, realizado en Puerto Vallarta, México* del 25 al 27 de Marzo de 2015. Finalmente, se participó en el evento internacional *II CONGRESO INTERNACIONAL DE ROBÓTICA Y COMPUTACIÓN*, realizado en Los Cabos, B.C.S., México, del 28 al 30 de Abril de 2015, en la modalidad de ponencia y se incluyó el artículo arbitrado en las memorias del evento del trabajo titulado "Generación de rutas de exploración utilizando un generador de números aleatorios caótico".

Por otro lado, se muestran las evidencias de la estancia de investigación que se realizó en el Departamento de Ingeniería de la Universidad de Ferrara, Italia. El proyecto de investigación se llevó a cabo bajo la asesoría y supervisión del Dr. Gianluca Setti y el Dr. Fabio Pareschi, especialistas en el tema.

E.1. Publicaciones

Análisis de la simetría de un oscilador caótico basado en un algoritmo numérico de orden adaptivo

C.H. Pimentel-Romero, J.L. Bueno-Ruíz, R. Huerta-Barrera, J.M. Muñoz-Pacheco, O.G. Félix-Beltrán, L.C. Gómez-Pavón, A. Luis-Ramos

jesusm.pacheco@correo.buap.mx

Facultad de Ciencias de la Electrónica, Benemérita Universidad Autónoma de Puebla,
Av. San Claudio y 18 Sur S/N Col. Jardines de San Manuel, Puebla, Pue., México.

RESUMEN

En este trabajo se presenta el análisis de simetría en el plano de fase de un oscilador caótico de cuatro enrollamientos basado en una función saturada. El sistema caótico se representa mediante variables de estado y su solución numérica se realiza mediante aproximaciones basadas en el método numérico multipasos Adams-Moulton en un esquema de orden variable. Se verifica la simetría del atractor cuantificando la evolución de la trayectoria caótica alrededor de sus puntos de equilibrio lo que permite evaluar el impacto de las variaciones de los parámetros del sistema caótico. El cálculo de error de truncamiento en cada iteración determina el orden y el tamaño de paso de integración con el que se soluciona el sistema caótico. Resultados de simulación numérica en MATLAB demuestran la aplicación del análisis propuesto.

PALABRAS CLAVE: Atractor, Función Saturada, Método Numérico, Multipasos, Oscilador Caótico.

1 INTRODUCCIÓN

Durante más de una década se han investigado los sistemas dinámicos no lineales con comportamiento caótico para utilizarse en una variedad aplicaciones en diversos campos, como en las matemáticas, la física, la ingeniería, la economía, la sociología, etc. [1-7]. Los sistemas caóticos poseen comportamientos dinámicos complejos, es decir son sistemas deterministas a largo término, su comportamiento es difícil de distinguir de un sistema aleatorio debido a que su espectro de frecuencia es similar al del ruido blanco dentro de su ancho de banda, y son extremadamente sensibles a las variaciones en los valores de sus parámetros y de las condiciones iniciales [8]. Cuantitativamente el exponente máximo de Lyapunov es un signo definitorio de caos mientras este tenga una magnitud positiva.

Las aplicaciones prácticas de los sistemas caóticos únicamente son posibles mediante la síntesis a nivel circuito electrónico, es decir pasar de una representación de variables de estado a un representación a nivel circuito y su realización experimental. Sin embargo, aunque existen diversas técnicas para diseñar osciladores caóticos empleando una diversidad de dispositivos electrónicos tales como diodos, amplificadores operacionales de voltaje, amplificadores operacionales de transconductancia, incluso circuitos integrados; estos sistemas dependen de procesos de diseño a la medida. Esto significa que cualquier cambio en las especificaciones del diseño repercute en un proceso de rediseño completo. Adicionalmente, la distribución uniforme en el espacio de fase de las trayectorias del sistema caótico son necesarias para validar diseños de atractores caóticos de múltiples enrollamientos, los cuales se han visualizado como posibles candidatos para incrementar las características de los sistemas caóticos y por ende sus aplicaciones. Por ejemplo se ha demostrado que atractores caóticos de múltiples enrollamientos pueden incrementar la caoticidad

CCADET
CENTRO DE CIENCIAS APLICADAS Y DESARROLLO TECNOLOGICO

SOMIXXIX
CONGRESO DE INSTRUMENTACIÓN
Puerto Vallarta, Jalisco, México, del 29 al 31 de octubre del 2014

SOMI

UNIVERSIDAD DE GUADALAJARA
Centro Universitario de la Costa

La Sociedad Mexicana de Instrumentación, el Centro de Ciencias Aplicadas y Desarrollo Tecnológico
de la Universidad Nacional Autónoma de México y la Universidad de Guadalajara,
a través del Centro Universitario de la Costa

Otorgan la presente

CONSTANCIA

a: Jesús Manuel Muñoz Pacheco, César Hugo Pimentel Romero, José Luis Bueno Ruiz, Ronald Huerta Barrera,
Olga Gudalupe Félix Beltrán, Luz del Carmen Gómez Pavón, Arnulfo Luis Ramos

por haber presentado su trabajo

Análisis de la simetría de un oscilador caótico basado en un algoritmo numérico de orden adaptivo

Puerto Vallarta, Jalisco, México, 31 de octubre del 2014

Por el Comité Organizador

M.I. Gerardo A. Ruíz Botello

Dr. Jorge I. Chavoya Gama

Fault conditions of a simple chaotic circuit under capacitor nonlinear effects

Special Session: Issues in electronic design automation: Tolerance analysis and design verification

J.L. Bueno-Ruiz, C.A. Arriaga-Arriaga, G.V. Cruz-Domínguez, C.H. Pimentel-Romero, J.M. Muñoz-Pacheco*, L.C. Gómez-Pavón, O. Félix-Beltrán, A. Luis-Ramos
Facultad de Ciencias de la Electrónica,
Benemérita Universidad Autónoma de Puebla, Puebla, MEXICO
*Email: jesusm.pacheco@correo.buap.mx

***Abstract*—In this paper a tolerance analysis in the electronic design of a simple chaos generator is reported. This simple chaotic oscillator is composed by four resistors, three capacitors and two opamps. A Verilog-A model for the opamps and capacitors is used herein. For the opamp, the model contains input impedance, finite bandwidth with a dominant pole and voltage saturation effects. In case of capacitor, a nonlinear model based on a varactor is considered, which includes the charge-dependence with the voltage. By using H-Spice simulator, the sensitivity of the chaos generation in the simple chaotic oscillator as a function of the varactor is analyzed. Several H-Spice simulations are given.**

***Index Terms*—Chaos, Oscillator, Verilog, Nonlinear Capacitor, Varactor, Modelling.**

I. Introduction

Since early 90's, with the work of Caroll and Pecora, was demonstrated that chaotic behavior can be controlled [1]. Therefore, in the last two decades, circuit implementation of various chaos generators has been of increasing interest, specially by their applications in engineering like sigma-delta modulators, random number generators, mobile robots, motor drivers of electric vehicles, biological systems, secure communications, etc. [2]. In this framework, a tendency is to design chaotic oscillators with a reduced form factor, e.g., Piper and Sprott [3] propose a simple chaotic circuit composed by four resistors, three capacitors and only two opamps.

It is well known that chaotic behavior is extremely sensitive to small variations of its parameter and initial conditions. By one side, variations in initial conditions conduct to multiple trajectories in time domain, but all converge to the same chaotic attractor in phase space domain. However, small changes in its parameters can lead to different dynamical behaviors in both domains, even, lose the chaotic behavior [4]-[6].

In literature, the design of chaotic oscillators can be classified as custom-made designs due to their parameters were tuned for a specific application, and any deviation of these, has important issues such as multiple re-design cycles or to destroy the chaotic regime [1]-[6]. So that, it is priority to gain insight about the tolerances that causes fault conditions in chaotic oscillators.

Therefore, this paper presents a tolerance analysis of the system parameters in the Sprott's simple chaotic oscillator [3]. Verilog-A is used herein to capture the high-level analog behaviors focused in the second order effects of the circuit elements [7]-[9]. First of all, the behavioral model for the opamps includes input and output impedances, dominant pole, gain-bandwidth product and saturation effects. Besides, for the capacitors incorporates a nonlinear model based on a varactor, which includes the charge-dependence with the voltage. In this manner, the nonlinear capacitor model allow us to make a swept on an specific voltage region in order to find the tolerances for the chaotic oscillator as a funtion of the maximun capacitance change from nominal and the voltage change for maximun capacitance. Verilog-A models are embedded into the H-SPICE circuit simulator to perform a tolerance analysis of the aforementioned cases.

The paper is organized as follows. The chaotic oscillator is given in section II. Tolerance analysis by considering a varactor is reported in section III. Finally, conclusions are summary in section IV.

II. A Simple Chaotic Oscillator

The model of the simple chaotic oscillator proposed by Piper and Sprott can be written in terms of a state-variable equations system [3]. In this context, the chaotic system is given as follows:

$$\begin{aligned} \dot{x} &= y, \\ \dot{y} &= z, \\ \dot{z} &= -Cx - By - Az + C\text{sgn}(x). \end{aligned} \tag{1}$$

System (1) can be considered as a piecewise linear system with three linear regions. For outer regions, where the state variable x is far away from the equilibrium point located in the origen, the function *signum* is a constant. Meanwhile, the inner region connects the two outer regions with

$$\text{sgn}(x) - x \approx x/\epsilon, \quad |x| < \epsilon. \tag{2}$$

Consequently, chaotic system in (1) is transformed in

$$\begin{aligned} \dot{x} &= y, \\ \dot{y} &= z, \\ \dot{z} &= (C/\epsilon)x - By - Az, \end{aligned} \tag{3}$$

due to the constant term disappears. That chaotic system can generate a chaotic attractor with two scrolls. From point of

Generación de rutas de exploración utilizando un generador de números aleatorios caótico

C.H. Pimentel-Romero*, J. M. Muñoz-Pacheco, O. G. Félix-Beltrán, L. C. Gómez-Pavón
Facultad de Ciencias de la Electrónica BUAP, Puebla, Pue.
*Email: ceshugo.pro@gmail.com

***Resumen*—En este artículo se estudian verdaderos generadores de números aleatorios (TRNG) basados en sistemas dinámicos no lineales con comportamiento caótico para generar rutas de exploración de un robot autónomo. Para esto se propone el análisis del sistema caótico de dos enrollamientos del sistema de Chua, el cual es un sistema caótico basado en una función lineal a tramos (PWL). Posteriormente se diseña un TRNG con el oscilador caótico de Chua y se demuestra que es requerido un post procesamiento (técnica de Von Neumann) para optimizar la aleatoriedad en la generación de bits y así obtener un porcentaje de cobertura mayor en el área establecida.**

***Keywords*—Caos, criptografía, generador de números aleatorios (RNG), robots autónomos, planeación de rutas, circuito de Chua.**

I. Introduccción

Durante más de una década se han investigado los sistemas dinámicos no lineales con comportamiento caótico para utilizarse en una variedad de aplicaciones en diversos campos, como lo son las matemáticas, la física, la ingeniería, la economía, la sociología, etcétera [1–7].

Los sistemas caóticos son sistemas dinámicos extremadamente sensibles a las variaciones en las condiciones iniciales, es decir, pequeñas variaciones en dichas condiciones iniciales implican grandes diferencias en el comportamiento futuro. Además, el comportamiento de estos sistemas es difícil de distinguir de un sistema aleatorio, lo que los convierte en sistemas difíciles de predecir [8].

Por otro lado, el diseño de robots autónomos es vital para la exploración de espacios reducidos y peligrosos para la intervención humana [3]. Existen muchas aplicaciones en las cuales se necesita cubrir grandes áreas, que van desde la exploración de túneles en excavaciones arqueológicas, exploración de planetas, detección de minas en misiones militares, o robots autónomos más simples como aspiradoras, cortadoras de césped o incluso juguetes [2, 5, 6].

Es así como surge la idea del diseño de robots autónomos que describan trayectorias óptimas para realizar tareas de exploración buscando maximizar la eficiencia de búsqueda, mediante las características antes mencionadas de la señal caótica [1–7].

En robótica, el primer robot móvil que pudo seguir una trayectoria caótica fue propuesto por Nakamura y Seikikuchi, donde fue usada la ecuación de Arnold para generar los movimientos deseados [5]. Investigaciones posteriores acerca de la generación de trayectorias caóticas usando la ecuación de Lorenz, Hamilton, Van der Pol, Chua y basada en una función saturada se muestran en [1, 2, 5–7]. Sin embargo, el estudio de otros generadores caóticos para la eficacia de las trayectorias es un problema abierto.

Debido a que la característica más importante que determina el éxito de estos sistemas robóticos es el método para planear las trayectorias, la investigación se enfoca en proponer técnicas para generar rutas de exploración que permita analizar la eficiencia y la aleatoriedad en la exploración del espacio de búsqueda.

Existen básicamente dos tipos de generadores utilizados para producir secuencias aleatorias: Generadores de números aleatorios (RNGs[1]) y generadores de números pseudoaleatorios (PRNGs[2]). Los generadores RNG utilizan generalmente una fuente no determinista (fuente de entropía) junto con alguna función de procesamiento para producir aleatoriedad. La fuente de entropía consiste típicamente de alguna cantidad física, como el ruido en un circuito eléctrico, procesos de interrupción por parte del usuario (por ejemplo, pulsaciones de teclas o movimientos del mouse), los efectos cuánticos en un semiconductor o utilizando diversas combinaciones de las entradas antes mencionadas. Un PRNG utiliza una o más entradas y genera múltiples números "pseudoaleatorios". A las entradas de un PRNG 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 RNG. Los números aleatorios y pseudoaleatorios generados para aplicaciones criptográficas deben ser aleatorios e impredecibles [9].

Con base en lo anterior, en este trabajo de investigación se presenta el diseño de un generador de números aleatorios basado en el sistema caótico de Chua. En este caso, para incrementar la eficiencia del generador y evitar la generación de bits repetidos se utiliza la técnica Von Neumann. Como resultado, el generador de bits aleatorios incrementa su porcentaje de cobertura en un espacio definido de 30 x 30 unidades normalizadas.

II. Circuito de Chua

El circuito de Chua consta de cinco elementos: una resistencia lineal, un inductor, dos capacitores y una resistencia no lineal conocida como el diodo de Chua, (ver figura 1). Este puede ser modelado por medio de variables de estado de la siguiente forma:

[1]RNGs por sus siglas en inglés: random number generators
[2]PRNGs por sus siglas en inglés: pseudorandom number generators

SEP
SECRETARÍA DE EDUCACIÓN PÚBLICA

El Instituto Tecnológico de La Paz

Otorga el presente

Reconocimiento

A

César Hugo Pimentel Romero, Jesús Manuel Muñoz Pacheco, Olga Guadalupe Félix Beltrán y Luz del Carmen Gómez Pavón

Por su artículo: Generación de rutas de exploración utilizando un generador de números aleatorios caótico
Presentado como ponencia en el
Segundo Congreso Internacional de Robótica y Computación
Celebrado del 28 al 30 de abril del año en curso
Y publicado con ISBN: 978-607-95534-8-7

La Paz, B.C.S., 30 de abril de 2015.

SECRETARÍA DE EDUCACIÓN PÚBLICA
TECNOLÓGICO NACIONAL DE MÉXICO
INSTITUTO TECNOLÓGICO DE LA PAZ
DIRECCIÓN

Ing. Jesús David Estrada Ruíz
Director

75

E.2. Estancia de Investigación

MÉXICO
PRESIDENCIA DE LA REPÚBLICA

CONACYT
Consejo Nacional de Ciencia y Tecnología

Posgraduate AND Scholarships Office

Scholarships Office

Mexico City, February 24th 2015.

Scholar reference: **553588**

TO WHOM IT MAY CONCERN:

El Consejo Nacional de Ciencia y Tecnologia - CONACYT (The National Council FOR Science AND Technology) certifies BY this document that **PIMENTEL ROMERO CESAR HUGO**, WITH register **553588** has a scholarship AS visiting research student AT **UNIVERSITA DEGLI STUDI DI FERRARA**, **ITALY**, FROM **April 2015** to **June 2015**.

This student will receive a monthly support for $15,000.00 Mexican Pesos, $670.00 Mexican Pesos monthly FOR medical insurance and $6,000.00 Mexican Pesos FOR air ticket.

Sincerely yours,

Mtro. Pablo Rojo
Scholarships Director.

DIPARTIMENTO DI INGEGNERIA
UNIVERSITA' DEGLI STUDI DI FERRARA
Via Saragat, 1 - 44122 FERRARA
Tel. 0532 / 974924 - Fax 0532 / 974870

To the attention of:

National Council of Science and Technology
(CONACYT)
Av. Insurgentes Sur 1582, Col. Credito Constructor
Del. Benito Juarez, Mexico, D.F.
C.P.: 03940, MEXICO.

From:

Professor Gianluca Setti, PhD
Engineering Department (ENDIF)
University of Ferrara
Via Saragat, 1
44122 Ferrara, ITALY
Tel: (39) 0532 97 4997
Fax: (39) 053297 4870
e-mail: gianluca.setti@unife.it

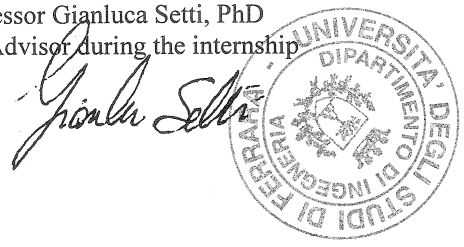
Ferrara, September 8, 2015

To whom it may concerns

With this letter I would inform you that Mr. Cesar Hugo Pimentel-Romero, ID:213470723, alumnus of the Facultad de Ciencias de la Electrónica at Benemérita Universidad Autónoma de Puebla (BUAP), Mexico, fulfilled satisfactorily the objectives established in the work plan of the academic internship under my co-advice from April 1th, 2015 to June 30, 2015 in the Department of Engineering (ENDIF) of the University of Ferrara, Ferrara, Italy.

Please, do not hesitate contact me for further information.

Best regards.
Professor Gianluca Setti, PhD
Co-Advisor during the internship



Bibliografía

[1] A. Sekiguchi and Y. Nakamura. The chaotic mobile robot. In *Intelligent Robots and Systems, 1999. IROS '99. Proceedings. 1999 IEEE/RSJ International Conference on*, volume 1, 1999.

[2] Xibo Wang, Li Ma, and Xiaozhou Du. An encryption method based on dual-chaos system. In *Intelligent Networks and Intelligent Systems, 2009. ICINIS '09. Second International Conference on*, pages 217–220, Nov 2009. doi: 10.1109/ICINIS.2009.63.

[3] Heung-Gyoon Ryu and Jun-Hyun Lee. High security wireless cdsk-based chaos communication with new chaos map. In *Military Communications Conference, MILCOM 2013 - 2013 IEEE*, pages 786–790, Nov 2013. doi: 10.1109/MILCOM.2013.139.

[4] Chin-Feng Lin, Shun-Han Shih, Jin-De Zhu, and Sang-Hung Lee. Implementation of an offline chaos-based eeg encryption software. In *Advanced Communication Technology (ICACT), 2012 14th International Conference on*, pages 430–433, Feb 2012.

[5] S. Arita. Relationship between logistic chaos and randomness using a matrix of probabilities and its application to the classification of time series: The line from chaos point to the reversed type of chaos is perpendicular to that from randomness's point to its reversed type. In *Soft Computing and Intelligent Systems (SCIS), 2014 Joint 7th International Conference on and Advanced Intelligent Systems (ISIS), 15th International Symposium on*, pages 1021–1028, Dec 2014. doi: 10.1109/SCIS-ISIS.2014.7044908.

[6] Shui Ying Xiang, Wei Pan, Bin Luo, Lian Shan Yan, Xi Hua Zou, Ning Jiang, Lei Yang, and Nian Qiang Li. Synchronization of unpredictability-enhanced chaos in vcsels with variable-polarization optical feedback. *Quantum Electronics, IEEE Journal of*, 47(10): 1354–1361, Oct 2011. ISSN 0018-9197. doi: 10.1109/JQE.2011.2166536.

[7] A.J. Lawrance. Recent theory and new applications in chaos communications. *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, pages 2446–2449, 2010.

[8] R.M. Nguimdo, R. Lavrov, P. Colet, M. Jacquot, Y.K. Chembo, and L. Larger. Effect of fiber dispersion on broadband chaos communications implemented by electro-optic nonlinear delay phase dynamics. *Lightwave Technology, Journal of*, 28(18):2688–2696, 2010.

[9] X. Xu, J. Guo, and H. Leung. Blind equalization for power-line communications using chaos. *Power Delivery, IEEE Transactions on*, (99):1–8, 2013.

[10] Chaowen Shen, Simin Yu, Jinhu Lu, and Guanrong Chen. A systematic methodology for constructing hyperchaotic systems with multiple positive lyapunov exponents and circuit implementation. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 61(3): 854–864, 2014.

[11] Steven H. Strogatz. *Nonlinear Dynamics and Chaos: with applications to physics, biology, chemistry and engineering*. Perseus Books, 1994.

[12] Q. Jia and X. Wang. Path planning of mobile robots in dynamic environment using chaotic prediction. In *Control and Decision Conference, 2008. CCDC 2008. Chinese*, pages 925–930, 2008.

[13] Y. Bae. Target searching method in the chaotic mobile robot. In *Digital Avionics Systems Conference, 2004. DASC 04. The 23rd*, volume 2, pages 12.D.7–12.1–9 Vol.2, 2004.

[14] P. Sooraska and K. Klomkarn. "no-cpu" chaotic robots: From classroom to commerce. *Circuits and Systems Magazine, IEEE*, 10(1):46–53, 2010.

[15] Fei Xia, Louae Tyoan, Zhongtao Yang, I. Uzoije, Guangcong Zhang, and P.A. Vela. Human-aware mobile robot exploration and motion planner. In *SoutheastCon 2015*, pages 1–4, April 2015. doi: 10.1109/SECON.2015.7133021.

[16] Jinhu Lu and Guanrong Chen. A brief overview of multi-scroll chaotic attractors generation. In *Circuits and Systems (ISCAS). Proceedings. 2006 IEEE International Symposium on*, number 4, pages 702–705, 2006.

[17] Lidong Wang, Xiuying Xing, and Zhenyan Chu. On definitions of chaos in discrete dynamical system. In *Young Computer Scientists, 2008. ICYCS 2008. The 9th International Conference for*, pages 2874–2878, Nov 2008. doi: 10.1109/ICYCS.2008.296.

[18] J. Palacin, J.A. Salse, I. Valganon, and X. Clua. Building a mobile robot for a floor-cleaning operation in domestic environments. *Instrumentation and Measurement, IEEE Transactions on*, 53(5):1418–1424, 2004.

[19] Ch. K. Volos, I.M. Kyprianidis, and I. N. Stouboulos. A chaotic path planning generator for autonomous mobile robots. *Robotics and Autonomous Systems*, 60(4):651 – 656, 2012.

[20] J. Nechvatal M. Smid E. Barker S. Leigh M. Levenson M. Vangel D. Banks A. Heckert J. Dray S. Vo A. Rukhin, J. Soto. A statistical test suite for random and pseudorandom number generators for cryptographic applications. *NIST National Institute of Standards and Technology*, Special Publication 800-22 Revision 1a, April2010.

[21] Massimo Cencini, Fabio Cecconi, and Angelo Vulpiani. *Chaos From Simple Models to Complex Systems*, volume 17. World Scientific Publishing, 2010.

[22] J.M. Muñoz-Pacheco and E. Tlelo-Cuautle. *Electronic design automation of multi-scroll chaos generators*. Bentham Sciences Publishers, 2010.

[23] J. G. Barajas-Ramírez. Multiple stable attractors in pwl chaotic systems. *Congreso Nacional de Control Automático (CNCA)*, pages 454–458, 2013.

[24] I. Mayoral-Juárez, J.M. Muñoz Pacheco, Félix-Beltrán, L.C. O. Gómez-Pavón, and A. Luis-Ramos. Determining the number of scrolls in a multi-scroll chaotic oscillator under uncertainties. *Power, Electronics and Computing (ROPEC), 2013 IEEE International Autumn Meeting on*, 2013.

[25] Ana Dalia, Pano Azucena, Esteban Tlelo Cuautle, Mauro Sánchez Sánchez, Luis Abraham, and Sánchez Gaspariano. Diseño de osciladores de múltiples enrollamientos combinando matlab y spice. *1er Congreso Iberoamericano de Instrumentación y Ciencias Aplicadas*, 2013.

[26] P. A. Cook. *Nonlinear Dynamical Systems*. Prentice Hall, 1994.

[27] J.M. Cruz and L.O. Chua. A cmos ic nonlinear resistor for chua ' s circuit. *IEEE Trans. Circuits Syst. I*, 1992.

[28] E. Tlelo-Cuautle, A. Gaona-Hernández, and J. García-Delgado. Implementation of a chaotic oscillator by designing chua ' s diode with cmos cfoas. *Analog Integrated Circuits and Signal Processing*, 48(2):159 – 162, 2006.

[29] Raymond P. Canale Steven C. Chapra. *Métodos Numéricos para Ingenieros*. 5 edition, 2006.

[30] A.A. Andronov, E.A. Vitt, and S.E. Khaikin. *Theory of Oscillations*. Pergamon Press, 1966.

[31] Vadim S. Anishchenko, Vladimir Astakhov, Tatjana Vadivasova, Alexander Neiman, and Lutz Schimansky-Geier. *Dynamical Chaos-Models and Experiments*. Springer Series in Synergetics. World Scientific, Singapore, 1995.

[32] F. Pareschi, G. Setti, and R. Rovatti. Implementation and testing of high-speed cmos true random number generators based on chaotic systems. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 57(12):3124–3137, Dec 2010.

[33] M.E. Yalcin, J. A K Suykens, and J Vandewalle. True random bit generation from a double-scroll attractor. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 51(7):1395–1404, July 2004.

[34] Michael T. Rosenstein, James J. Collins, Carlo J. De Luca, and Corresponding Michael. A practical method for calculating largest lyapunov exponents from small data sets. *Physica D*, 65:117–134, 1993.