UNIVERSIDAD CATOLICA
DE SANTIAGO DE GUAYAQUIL

SISTEMA DE POSGRADO

MAESTRIA EN TELECOMUNICACIONES

TEMA:

Estudio de la sincronización de sistemas caóticos para garantizar la seguridad de las comunicaciones

AUTOR:

Tutivén Gálvez, Christian Javier

Trabajo de titulación previo a la obtención del grado de

MAGÍSTER EN TELECOMUNICACIONES

Tutor:

Ing. Manuel Romero Paz, Msc.

Guayaquil, 18 de mayo de 2018

UNIVERSIDAD CATOLICA
DE SANTIAGO DE GUAYAQUIL

SISTEMA DE POSGRADO

MAESTRIA EN TELECOMUNICACIONES

CERTIFICACIÓN

Certificamos que el presente trabajo de titulación, fue realizado en su totalidad por **CHRISTIAN JAVIER TUTIVÉN GÁLVEZ**, como requerimiento para la obtención del Título de **MAGISTER EN TELECOMUNICACIONES**.

TUTOR

_____

ING. MANUEL ROMERO PAZ, MSC

DIRECTOR DEL PROGRAMA

_____

ING. MANUEL ROMERO PAZ, MSC

Guayaquil, 18 de mayo de 2018

UNIVERSIDAD CATOLICA
DE SANTIAGO DE GUAYAQUIL

SISTEMA DE POSGRADO

MAESTRIA EN TELECOMUNICACIONES

DECLARACIÓN DE RESPONSABILIDAD

Yo, **CHRISTIAN JAVIER TUTIVÉN GÁLVEZ**

DECLARO QUE:

El Trabajo de Titulación **"ESTUDIO DE LA SINCRONIZACIÓN DE SISTEMAS CAÓTICOS PARA GARANTIZAR LA SEGURIDAD DE LAS COMUNICACIONES"** previa a la obtención del Título de **MASTER EN TELECOMUNICACIONES**, ha sido desarrollado en base a una investigación exhaustiva, respetando derechos intelectuales de terceros conforme las citas que constan al pie de las páginas correspondientes, cuyas fuentes se incorporan en la bibliografía. Consecuentemente este trabajo es de mi total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance científico del Trabajo de Titulación referido.

Guayaquil, 18 de mayo de 2018

EL AUTOR

_____

CHRISTIAN JAVIER TUTIVÉN GÁLVEZ

UNIVERSIDAD CATOLICA
DE SANTIAGO DE GUAYAQUIL

SISTEMA DE POSGRADO

MAESTRIA EN TELECOMUNICACIONES

AUTORIZACIÓN

Yo, **CHRISTIAN JAVIER TUTIVÉN GÁLVEZ**

Autorizo a la Universidad Católica de Santiago de Guayaquil, la **publicación** en la biblioteca de la institución del Trabajo de Titulación: **"ESTUDIO DE LA SINCRONIZACIÓN DE SISTEMAS CAÓTICOS PARA GARANTIZAR LA SEGURIDAD DE LAS COMUNICACIONES"**, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y total autoría.

Guayaquil, 18 de mayo de 2018

EL AUTOR:

_____

CHRISTIAN JAVIER TUTIVÉN GÁLVEZ

DEDICATORIA

En primer lugar a Dios le doy gracias a Dios por darme la vida y la sabiduría para poder culminar una nueva etapa en mi formación profesional y personal, a mis padres porque siempre han sido mi apoyo, mi motor para el logro de metas y sobretodo mi ejemplo a seguir, a mis profesores de la Maestría en Telecomunicaciones por todos los conocimientos adquiridos gracias a ellos y que me sirvieron para el desarrollo de esta investigación y que me serán de utilidad a lo largo de mi vida profesional.

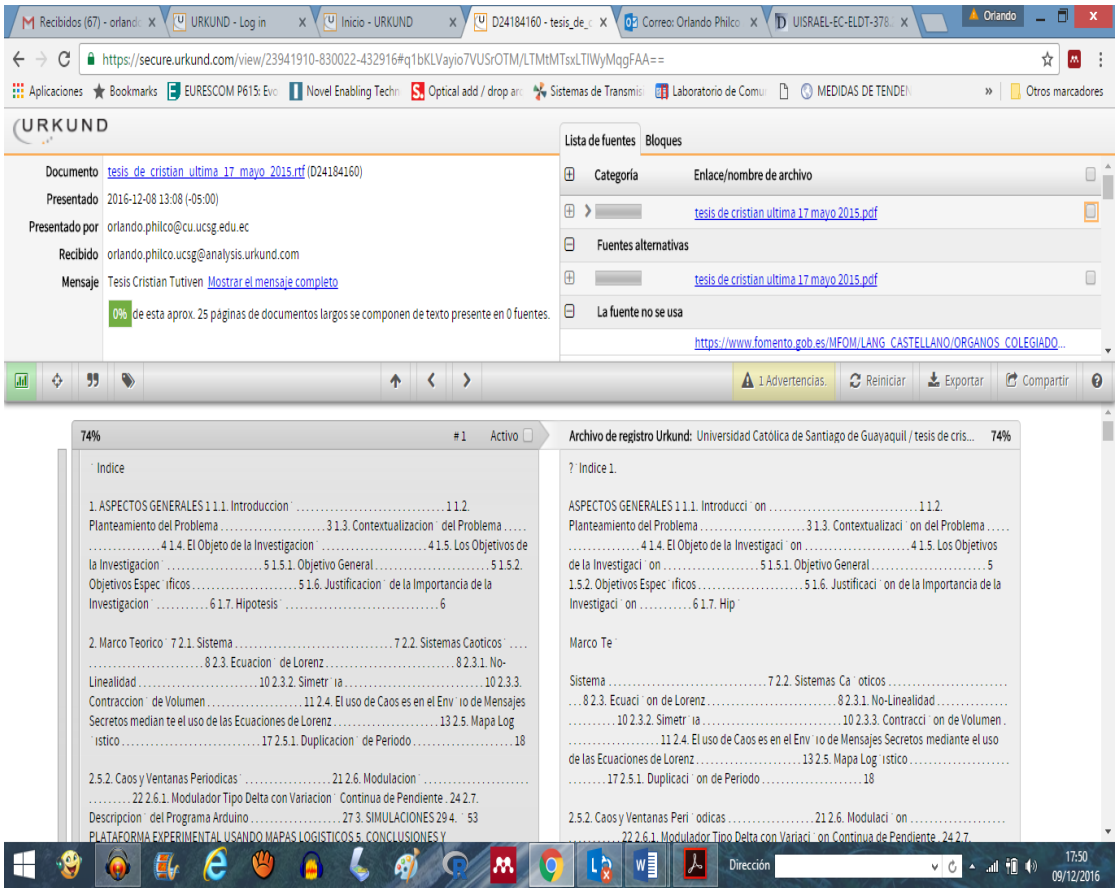
AGRADECIMIENTO

A mi familia, por el apoyo que me han brindado a lo largo de mi vida, porque siempre están pendientes de mi superación profesional y personal.

A los maestros que he tenido a lo largo de mi vida académica, porque gracias a cada uno de ellos he podido culminar esta maestría y podré conseguir muchas metas a lo largo de mi vida.

Al Ing. Manuel Romero, por su ayuda apoyo incondicional a lo largo de la maestría.

REPORTE URKUND



Reporte Urkund Tesis Maestría Telecomunicaciones: Estudio de la sincronización de sistemas caóticos para garantizar la seguridad de las comunicaciones del ing. Cristian Tutivén, al 0%

UNIVERSIDAD CATÓLICA

DE SANTIAGO DE GUAYAQUIL

SISTEMA DE POSGRADO

MAESTRIA EN TELECOMUNICACIONES

TRIBUNAL DE SUSTENTACIÓN

f. _____

Romero Paz, Manuel MSc.

TUTOR

f. _____

Córdova Rivadeneira, Luis MSc.

REVISOR

f. _____

Philco Asqui, Orlando MSc.

REVISOR

f. _____

Romero Paz, Manuel MSc.

DIRECTOR DEL PROGRAMA

RESUMEN

Todos los días se realizan millones de comunicaciones y la información se ha convertido en un activo de gran valor para las personas y las instituciones. Por lo que garantizar la seguridad de la información que se transmite es de vital importancia para así no sufrir pérdidas ni robos de la misma que puedan causar daños económicos y de confianza.

Muchos métodos se han investigado y desarrollado a lo largo de los años pero la mayoría usan técnicas que usan modelos que son fácil de predecir y de decodificar. Por lo que el uso de técnicas que usen modelos matemáticos con comportamientos caóticos ha ganado mucho interés.

En esta tesis se presenta un estudio de las señales caóticas y su uso para garantizar la seguridad de las comunicaciones, además que se realiza un esquema de fácil implementación en un microprocesador para poder realizar pruebas y prácticas de laboratorio.

En el capítulo 1, se hace referencia a los aspectos generales de la investigación.

En el capítulo 2 se presenta un estudio de las señales caóticas, del sistema de modulación aplicado y del microprocesador usado en el esquema experimental.

En el capítulo 3, se realizan distintas simulaciones, usando el programa Mathematica para comprobar el comportamiento de las señales caóticas y del mapa logístico.

En el capítulo 4 se presenta, un esquema propuesto de fácil implementación del mapa logístico para garantizar la seguridad de la comunicación.

Finalmente, en el capítulo 5 se presentan conclusiones y recomendaciones de la investigación realizada.

PALABRAS CLAVES: Sistema, Sistemas Caóticos, Simetría, No-Linealidad, Modulación, Arduino

ABSTRACT

Every day millions of communications are performed and the information has become a valuable asset for individuals and institutions. So guaranteeing the security of the information that is transmitted is of vital importance so as not to suffer losses or theft of the it that can cause economic damages and of confidence.

Many methods have been researched and developed over the years but most models using techniques which are easily to predict and decoded. Thus techniques that used mathematical models with chaotic behaviors have gained much interest.

In this thesis, a study of chaotic signals and their use to ensure the security of communications is presented, in addition is presented a scheme of easy implementation in a microprocessor to perform tests and laboratory practices.

In Chapter 1, is made a reference about the general aspects of the research.

In Chapter 2 a study of chaotic signals, the applied modulation scheme and the microprocessor used in the experimental scheme is studied.

In Chapter 3, several simulations are performed using the Mathematica program to check the behavior of chaotic signals and the logistic map.

Chapter 4 presents a proposed scheme of easy implementation to ensure the security of the communication scheme.

Finally, in Chapter 5 Conclusions and recommendations of the research are presented.

KEYWORDS: System, Chaotic Systems, Symmetry, Non-Linearity, Modulation, Arduino

Índice

Índice de figuras

1. ASPECTOS GENERALES

1.1. Introducción

La seguridad y el secreto de la información son aspectos importantes en las comunicaciones a nivel mundial pero a lo largo de los años, las técnicas de encriptamiento estaban restringidas para el área militar. Con el paso del tiempo, debido al gran crecimiento a nivel mundial de las redes de comunicación y de los sistemas digitales, estas técnicas han extendido su uso en instituciones como bancos, empresas privadas, organizaciones educativas, etc.

En los últimos años, varias técnicas de encriptación se han propuesto para así mejorar la seguridad en la transmisión de la información. Técnicas de encriptación basadas en señales caóticas han tenido un estudio extenso como parte de la solución al problema, debido a la alta imprevisibilidad y al comportamiento aleatorio natural de las señales caóticas. Entre las investigaciones realizadas podemos nombrar, la modificación del oscilador tipo Chua y su aplicación para la seguridad de las comunicaciones [Zapateiro, 2014], métodos de sincronización adaptativa para señales de transmisión en portadores caóticos [Andrievsky, 2002] [Moez, 2003], un sistema de transmisión de señales digitales usando una sincronización caótica [Parlitz, 1992], una demostración experimental de la seguridad en las transmisiones usando sincronización caótica [Kocarev, 1992], la demostración de comunicaciones de alta velocidad en largas distancia basadas en sincronización caótica en canales de fibra óptica comercial [Argyris and Mirasso, 2005], un simple diseño de un observador adaptativo para la estimación de parámetros desconocidos de la transmisión basado en el diseño cuadrático de la función de Lyapunov

en el error del sistema [Fradkov, 2000], el uso de una sincronización proyectiva en el esquema de seguridad de una comunicación [Li, 2004] y el uso de un sistema de switching caótico para modulación paramétrica [Yang, 1996], etc. Esto demuestra el gran interés científico e investigativo en el uso de las señales caóticas en la transmisión de información.

Hay que tener en cuenta que existen 2 maneras para diseñar un sistema de seguridad basado en la dinámica de las señales caóticas: analógicas y digitales. Los sistemas de comunicación analógica basados en caos son posibles, debido a la posibilidad de sincronización [Pecora, 1989]. La sincronización ocurre cuando la salida del sistema master controla la respuesta del sistema esclavo haciendo que ambas oscilaciones estén sincronizadas de la misma manera. Los sistemas de comunicación digital basados en caos no depende de la sincronización en su totalidad. Lo que hace es usar un mapa caótico en el cual las condiciones iniciales y el control de parámetros juegan el rol de una llave secreta [Alvarez, 2006].

Así mismo, numerosos trabajos que usan un mapa logístico (mapa caótico) para mejorar la seguridad en la comunicación, se pueden encontrar en la literatura. El mapa logístico es un mapa discreto no lineal, el cual sera descrito en la sección 2.5. Entre los trabajos que se pueden nombrar están la presentación de una trabajo experimental de un esquema de encriptación caótica, usando a Xilin Vitex 6 FPGA [Pande, 2010], también el cifrado simétrico de texto en el cual se usa una llave secreta de 128 bits y dos mapas logísticos con secuencias pseudoaleatorias optimizadas [Murillo, 2014].

Así también se encuentra la implementación de un generador caótico de bits aleatorios el cual es implementado en un microcontrolador Arduino [Volos, 2013].

En este documento se estudiaran los sistemas caóticos y su sincronización para garantizar la seguridad de las comunicaciones, presentando distintos casos de estudios realizados. Además se presentará un esquema experimental del uso de señales caóticas para garantizar la seguridad de las comunicaciones usando un microcontrolador el cual puede ser de facíl implementación en los laboratorios de la universidad.

1.2. Planteamiento del Problema

A lo largo del tiempo se han estudiado diversas técnicas para garantizar la seguridad en las comunicaciones, pero también en paralelo se han desarrollado diversos métodos que permiten que agentes extraños (hackers) obtengan información de estas comunicaciones, sin tener alguna autorización. Muchas de las técnicas estudiadas o implantadas para garantizar la seguridad en las comunicaciones, son de fácil decodificación por estos agentes extraños, además de que por el desconocimiento de como implementar estás técnicas de codificación en un hardware, muchos prototipos de seguridad usan protecciones comunes y ya conocidas por los posibles intrusos.

1.3. Contextualización del Problema

Partiendo de lo general a lo particular sobre el estudio de un sistema que garantice la seguridad en las comunicaciones y que además sea de fácil implementación en los

laboratorios de la universidad para que los estudiantes puedan realizar practicas, se ha escogido el uso de señales caóticas y para el diseño del esquema de implementación, el uso de mapas logísticos que generen llaves de seguridad, por las propiedades que tienen estas señales y que las hacen de difícil predicción. Una de las desventajas es el desconocimiento de las propiedades de este tipo de señales y el rango óptimo de operación y útil para el diseño de sistemas de seguridad en comunicaciones, además del desconocimiento de como implementar este tipo de sistemas en un microprocesador.

1.4. El Objeto de la Investigación

El objeto de este trabajo es presentar las principales propiedades de las señales caóticas y sus distintos comportamientos a distintos rangos de configuración. Además de su posible uso en sistemas de comunicaciones como medio de seguridad de la información transmitida.

Este proyecto se enfocará básica y directamente en el análisis de las señales, mediante simulaciones en distintos programas (Mathematica y Simulink), y luego se presentará un diseño de una plataforma de seguridad de las comunicaciones usando un microprocesador Arduino el cual es de fácil implementación en los laboratorios de la universidad.

1.5. Los Objetivos de la Investigación

1.5.1. Objetivo General

Realizar un estudio de sistemas que garantice la seguridad en las comunicaciones usando señales caóticas que sean difíciles de descifrar debido a su extraño comportamiento, además de presentar un esquema experimental del uso de estas señales, implementadas en un microcontrolador que garantice la seguridad de la comunicación.

1.5.2. Objetivos Específicos

En virtud de alcanzar el objetivo principal de la investigación se han planteado 6 premisas fundamentales:

1. Repasar la teoría del funcionamiento de los sistemas.
2. Analizar el comportamiento de las señales caóticas y sus propiedades.
3. Estudiar diversos tipos de codificaciones mediante el uso de señales caóticas.
4. Realizar simulaciones del comportamiento de las señales para verificar su comportamiento aleatoria y beneficio de la garantía de la seguridad de las comunicaciones.
5. Estudiar distintos tipos de modulaciones.
6. Mostrar un esquema experimental usando señales caóticas.

1.6. Justificación de la Importancia de la Investigación

El haber escogido el uso de señales caóticas como medio de seguridad en las comunicaciones es sin duda la oportunidad de presentar una alternativa en las técnicas de encriptación usadas, además que permite estudiar el extraño comportamiento de estas señales que las hace especiales para su estudio e implementación en otros casos. Además el diseño un esquema que incorpore un microprocesador enriquece la investigación, ya que es un sistema que puede ser implementado por personas con conocimiento en telecomunicaciones como con conocimientos en control y en electricidad.

1.7. Hipótesis

La implementación de un esquema experimental usando señales caóticas permitirá poder realizar estudios de distintos sistemas de seguridad de la información y poder implementarlos de una manera sencilla en un microcontrolador para poder realizar pruebas reales, no simuladas.

1.8. Metodología

Para alcanzar los objetivos propuestos se seleccionaron los siguientes métodos de investigación:

- Método de análisis y síntesis.
- Método de la inducción y la deducción.
- Método de la observación.

1.9. Estructura de la tesis

En el **Capítulo 1** se presenta la concepción metodológica de la tesis de maestría: se definen los objetivos generales y específicos, los antecedentes, el planteamiento, justificación del problema y los métodos de investigación utilizados.

En el **Capítulo 2** se presenta el marco teórico de los sistemas caóticos y sus propiedades; así como, la descripción de la ecuación de Lorenz, las propiedades del mapa logístico, los tipos de modulación y demodulación y finalmente el microprocesador (Arduino) a usarse en el sistema de seguridad de las comunicaciones.

En el **Capítulo 3** se presentan el varias simulaciones del comportamiento de señales caóticas (mapa logístico) usando el programa Mathematica 6.0.3, DynPac .

En el **Capítulo 4** se muestra el desarrollo experimental del sistema de seguridad. Se presentan todos sus elementos y la debida configuración en el microcontrolador Arduino.

Finalmente, el **Capítulo 5** se exponen las conclusiones, recomendaciones y futuros trabajos a partir de esta tesis de la tesis.

2. Marco Teórico

2.1. Sistema

Un sistema es un conjunto de elementos o bloques que están interconectados entre sí para cumplir un objetivo deseado y que sin alguno de sus elementos no lo podría conseguir o para propósitos específicos también se lo puede definir como un modelo matemático que relaciona las señales de entrada (excitaciones) al sistema con sus señales de salida (respuestas).

Como se indico anteriormente, en los sistemas tenemos señales de entrada (a las cuales podremos llamar x) y señales de salida (que podemos llamar y). Si este es el caso, se puede representar esta notación como

$$y = T[x], \tag{2.1}$$

donde T es el operador que representa las operaciones realizadas para que la señal de excitación x sea transformada en la de respuesta y.

Como ejemplos de sistemas se puede observar la relación de la Ecu. 2.1 en la Fig. 2.1(a) para un sistema de una sola entrada y una sola salida y en la Fig. 2.1(b) para un sistema con varias entradas y varias salidas.

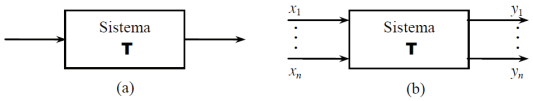


Figura 2.1: (a) Sistema de una sola entrada y una sola salida. (b) Sistema con entradas y salidas multiples.

Fuente:(el autor)

2.2. Sistemas Caóticos

Los sistemas pueden ser clasificados en 2 tipos [Moriello, 2015]:

- Sistema determinístico.- cuando su comportamiento es predecible, es decir que sigue ciertas reglas.
- Sistema estocástico.- cuando no hay seguridad de su estado futuro, sólo una probabilidad.

Uno de los sistemas dinámicos no lineales que se comporta, en ciertas condiciones, de una forma compleja que parece que sea de tipo estocástico, pero en realidad es determinístico, es el sistema caótico. Estos sistemas depende de las condiciones iniciales, y de hecho son muy sensibles a pequeños cambios en estas condiciones. Por ejemplo un pequeño cambio en las condiciones iniciales y sometiendo el sistema a influencias externas similares, producen respuestas finales totalmente distintas. Es por esto que es casi imposible realizar una predicción del estado final de estos sistemas.

2.3. Ecuación de Lorenz

Edward Lorenz(1963), deriva un sistema tridimensional a partir de un modelo de rollos de convección en la atmósfera, simplificado drásticamente. Sistema formado por las siguientes ecuaciones [Strogatz, 2000]:

$$\dot{x} = \sigma(y - x), \tag{2.2}$$

$$\dot{y} = rx - y - xz, \tag{2.3}$$

$$\dot{z} = xy - bz, \tag{2.4}$$

donde σ, r, $b > 0$ son parámetros.

Lorenz descubrió que este sistema determinista y que tiene un aspecto sencillo, podría tener una dinámica extremadamente erratica y que con un gran rango de parámetros las soluciones tienen oscilaciones irregulares, las cuales nunca se repiten exactamente igual pero todas se mantienen dentro de una región limitada de espacio-fase. Cuando Edward trazó las trayectorias en tres dimensiones, descubrió que estaban colocadas en un conjunto complicado ahora llamado atractor extraño . El atractor extraño no es un punto, curva ni superficie es un fractal con 2 y 3 dimensiones.

Entre las pruebas del comportamiento del sistema a largo plazo, que realizo Edward demostró que en cierto rango de parámetros, no puede haber puntos de equilibrio estables y no hay ciclos limites estables, sin embargo también demostró que todas las

trayectorias permanecen dentro de una región limitada y eventualmente son atraídas a un conjunto de volumen cero.

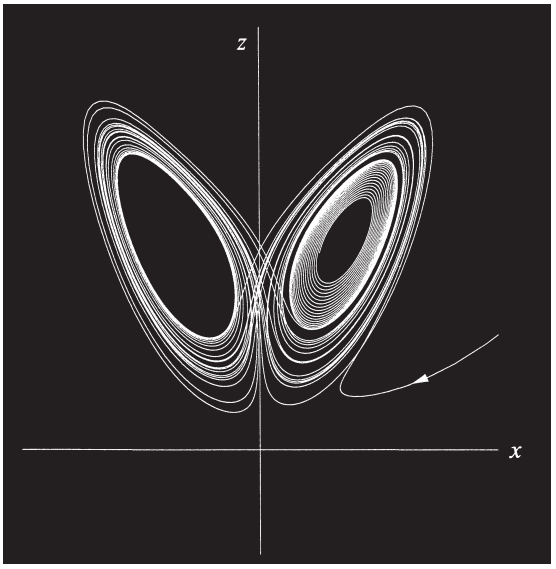


Figura 2.2: Imagen de trayectorias atraídas por un atractor extraño.

Fuente:([Strogatz, 2000])

Las ecuaciones de Lorenz tienen las siguientes propiedades:

2.3.1. No-Linealidad

El sistema de las ecuaciones (2.2), (2.3) y (2.4) tienen términos cuadráticos (xy y xz) que son no-linealidades del sistema.

2.3.2. Simetría

Si se reemplaza (x, y) por $(-x, -y)$ en el sistema de ecuaciones (2.2), (2.3) y (2.4), las ecuaciones se mantienen igual. Es decir si $(x(t), y(t), z(t))$ es una solución, también lo es $(-x(t), -y(t), z(t))$. Por lo que se puede deducir que todas las soluciones son

simétricas o tienen una pareja simétrica.

2.3.3. Contracción de Volumen

El sistema de Lorenz es disipativo, es decir que volúmenes en el espacio de fases se contraen bajo el flujo. Para demostrar esto se explicara el ejemplo mostrado en [Strogatz, 2000].

Para cualquier sistema de tres dimensiones $\dot{x} = f(x)$. Se escoge una superficie cerrada cualquiera $S(t)$ de volumen $V(t)$ en el espacio de fases. Asumiendo que s es una condición inicial para las trayectorias y permitiendo que evolucione a lo largo de un tiempo infinitesimal dt. Luego de esto S evoluciona a una nueva superficie $S(t + dt)$. Para conocer el nuevo volumen $V(t + dt)$ se realizo el siguiente análisis.

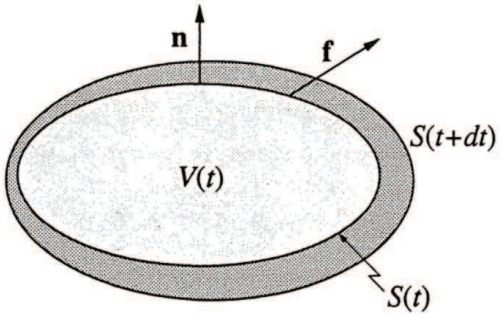


Figura 2.3: Vista lateral del volumen.

Fuente:([Strogatz, 2000])

Si **n** es la normal de salida de la superficie S, mientras **f** es la velocidad instantánea en un punto, entonces **f*n** es la componente normal de salida de la velocidad. Por lo tanto en el tiempo dt una parte de área dA produce un volumen (**f*n**dt)dA como se muestra

en la Fig. 2.4.

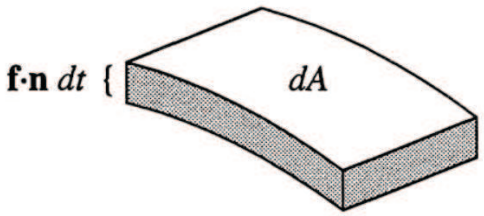


Figura 2.4: Vista del volumen $(f * ndt)dA$.

Fuente:([Strogatz, 2000])

Entonces

$$V(t+dt) = V(t) + (volumen \quad producido),$$

y se obtiene

$$V(t+dt) = V(t) + \int_s^t (\mathbf{f.n}dt)dA.$$

Entonces

$$\dot{V} = \frac{V(t+dt) - V(t)}{dt} = \int_s^t (\mathbf{f.n}dt)dA. \tag{2.5}$$

Se reescribe la ecuación bajo el teorema de divergencia, obteniendo

$$\nabla.f = \frac{\partial}{\partial x}[\sigma(y-x)] + \frac{\partial}{\partial y}[rx - y - xz] + \frac{\partial}{\partial z}[xy - bz] = -\sigma - 1 - b < 0.$$

Debido a que la divergencia es constante, la Ecu. 2.5 se reduce a $\dot{V} = -(\sigma + 1 + b)V$, cuya solución es $V(t) = V(0)e^{-(\sigma+1+b)t}$. Por lo que volúmenes en estado fase se encogen exponencialmente rápido.

2.4. El uso de Caos es en el Envío de Mensajes Secretos mediante el uso de las Ecuaciones de Lorenz

Una de la ventaja de los usos de los sistemas no lineales es que su propiedad caótica puede ser de mucha ayuda. De hecho en el año 1992 y 1993, Kevin Cuomo y Alan Oppenheim realizaron la implementación de un sistema para tener seguridad en las comunicaciones. Para esto descubrieron la sincronización del caos, cuya estrategia es que, cuando se transmite un mensaje, lo enmascara con un señal muy caótica [Cuomo, 1992] [Cuomo, 1993a] [Cuomo, 1993b]. Por lo cual, cualquier persona ajena que quiera escuchar el mensaje, escuchara solo la señal caótica. cuyo sonido sera parecido al ruido. Pero suponiendo que en el receptor puede reproducir el caos, el podrá substraer la mascara caótica y podrá escuchar el mensaje.

Para esto Kevin Cuomo muestra como pudo realizar una mascara caótica usando la implementación electrónica de la ecuación de Lorenz, la que se muestra en la Fig. 2.5 Este circuito esta formado por resistencias, capacitores, amplificadores operacionales y multiplicadores analógicos.

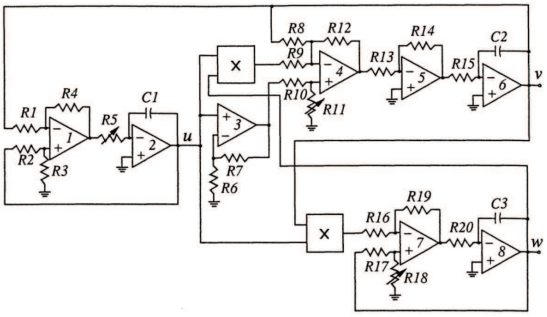


Figura 2.5: Implementación electrónica de la ecuación de Lorenz en el transmisor.

Fuente:([Cuomo, 1993a] [Cuomo, 1993b])

En el circuito de la Fig. 2.5 u, v y w son proporcionales a los valores de x, y y z en la ecuación de Lorenz.

Lo que buscaba Cuomo es lograr que el receptor pueda estar sincronizado perfectamente con el transmisor caótico. Para esto Cuomo creó un receptor (Fig. 2.6) idéntico al circuito de Lorenz y observo que existía una casi perfecta sincronización, aún así ambos circuitos estaban funcionando con sistemas caóticos, lo que se puede observar en la Fig., donde u_r y v_r siguen a sus contrapartes transmitidas $u(t)$ y $v(t)$.

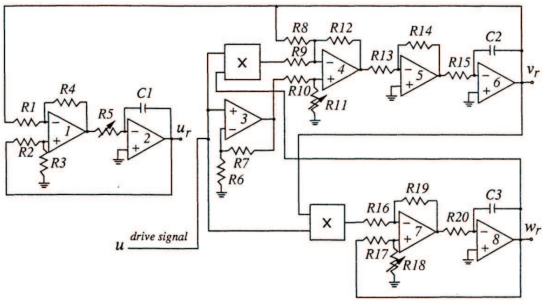


Figura 2.6: Implementación electrónica de la ecuación de Lorenz en el receptor.

Fuente:([Cuomo, 1993b])

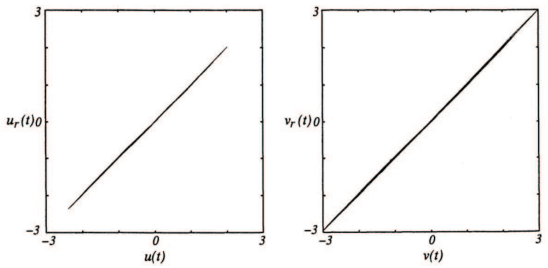


Figura 2.7: Relación entre las variables del receptor y del transmisor.

Fuente:([Cuomo, 1993b])

Una de las pruebas que realizo Cuomo fue la de enmascarar con una señal caótica, una parte de una canción y al tratar de escuchar el mensaje enmascarado, lo único que obtenía era ruido. Luego envío este mensaje a un receptor, cuya salida estaba perfectamente sincronizada con el caos original, con lo cual se pudo escuchar la parte de la canción. Las siguientes figuras muestran como fue realizada esta prueba y sus resultados. La Fig. 2.8 muestra el segmento de canción muestreada a una tasa de 48Hz. y con una resolución de 16 bits. Esta señal fue enmascarada con una señal muy caótica con una potencia 20 decibeles mayor que el mensaje como se muestra en la Fig. 2.10. Luego de desenmascarar el mensaje en el receptor se obtuvo la Fig. 2.9, donde se observa que el segmento de canción original fue recuperado con una pequeña distorsión.

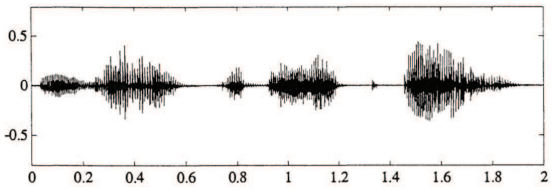


Figura 2.8: Muestra real del segmento de canción muestreada a una tasa de 48Hz. y una resolución de 16 bits.

Fuente:([Cuomo, 1993a] [Cuomo, 1993b])

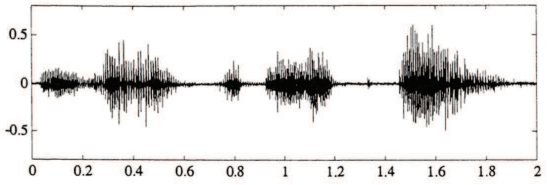


Figura 2.9: Muestra de señal obtenida luego de ser desenmascarada.

Fuente:([Cuomo, 1993a] [Cuomo, 1993b])

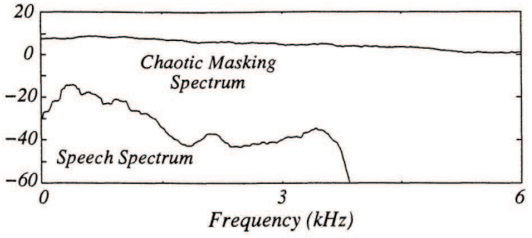


Figura 2.10: Espectro de potencia de segmento de canción y de la señal caótica.

Fuente:([Cuomo, 1993a] [Cuomo, 1993b])

Cuomo y Oppenheim continuaron sus estudios debido a que muchas personas dudaban de la sincronización de dos sistemas caóticos, ya que al ser sensibles a cualquier pequeño cambio en sus condiciones iniciales, podrían haber errores entre el transmisor

y el receptor y estos crecerían exponencialmente. Por lo cual presentaron la siguiente aclaración basados en la Fig. 2.6. El circuito del receptor es idéntico al del transmisor, excepto que la señal $u(t)$ reemplaza a la señal del receptor u_r en un lugar crucial del circuito.

Usando las ecuaciones de Kirchhoff obtuvieron:

$$\dot{u} = \sigma(v - u), \tag{2.6}$$

$$\dot{v} = ru - v - 20uw, \tag{2.7}$$

$$\dot{w} = 5uv - bw, \tag{2.8}$$

que representan la dinámica del transmisor. Así mismo la dinámica del receptor es

$$\dot{u}_r = \sigma(v_r - u_r), \tag{2.9}$$

$$\dot{v}_r = ru(t) - v_r - 20u(t)w_r, \tag{2.10}$$

$$\dot{w}_r = 5u(t)v_r - bw_r, \tag{2.11}$$

donde se observa que en la dinámica del receptor, que este sistema esta guiado por la señal caótica $u(t)$ que proviene del transmisor. Lo que muestra que el receptor se encuentra esta perfectamente sincronizado con el transmisor para cualquier condición inicial. Por lo que el error:

$$e = d - r \qquad y \qquad e(t) \to 0 \qquad cuando \qquad t \to \infty, \tag{2.12}$$

donde $d = (u, v, w)$ (los estados del transmisor) y $r = (u_r, v_r, w_r)$ (los estados del receptor).

2.5. Mapa Logístico

Existen muchas situaciones que pueden ser descritas mediante una simple ecuación diferencial de primer orden, donde estudiando las propiedades de la dinámica de sus modelos se pueden encontrar sus puntos de equilibrio y realizando un análisis de linealidad se puede determinar su estabilidad con respecto a pequeños disturbios. Pero en el año de 1976, Robert May publico un articulo donde indica que incluso mapas no lineales simples podrían tener una dinámica muy complicada. May demostró su punto mediante el mapa logístico:

$$x_{n+1} = r x_n (1 - x_n), \tag{2.13}$$

la versión discreta del modelo logístico de Verhulst [Verhulst, 1845], el cual se usaba para medir el crecimiento de la población, donde x_n es una medida adimensional de la población en la generación n y r es la tasa de crecimiento intrínseca. Como se muestra en la Fig. 2.11, al graficar la Ecu. (2.13), se obtiene una parabola con un valor máximo de $r/4$ en $x = 1/2$, para este caso, se dieron los valores de $r = 4$ y $0 \leq x_n \leq 1$.

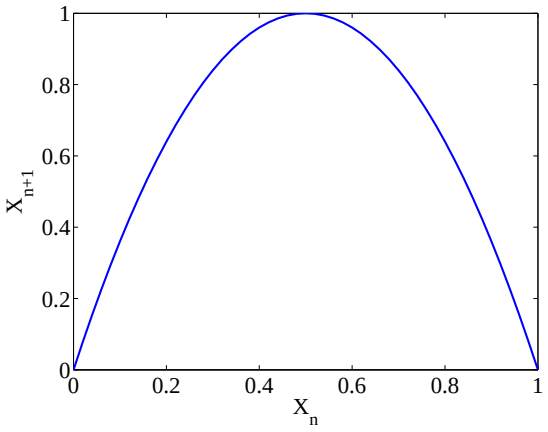


Figura 2.11: Resultado del mapa logístico con $0 \leq x_n \leq 1 \quad y \quad r = 4$.

Fuente:(el autor)

Entre las propiedades del mapa logístico tenemos:

2.5.1. Duplicación de Periodo

Si suponemos que r es un valor fijo y escogemos una población inicial de x_0, al usar la Equ. (2.13) para generar el siguiente x_n, tenemos las siguientes conclusiones (resultados obtenidos de las simulaciones realizadas en el programa Simulink, como se puede observar en la Fig. 2.12):

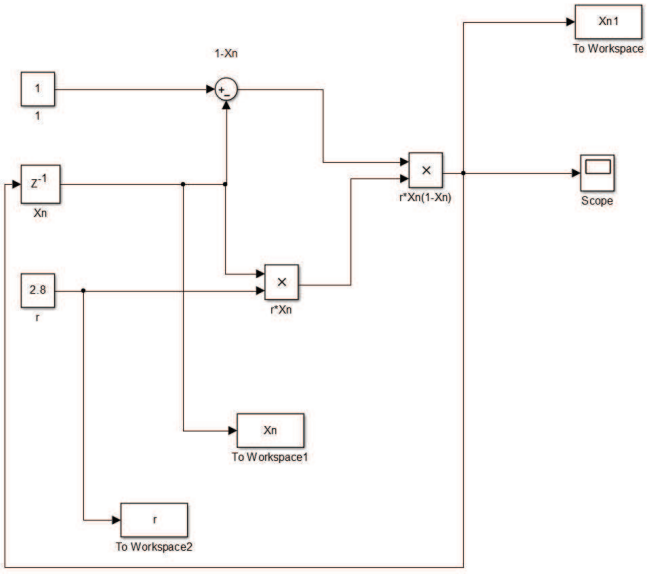


Figura 2.12: Resultado del mapa logístico con $0 \leq x_n \leq 1$ y $r = 4$.

Fuente:(el autor)

Para pequeñas tasas de crecimiento $r < 1$, la población siempre tiende a extinguirse, es decir $X_n \to 0$ cuando $n \to \infty$, como se muestra en la Fig. 2.13.

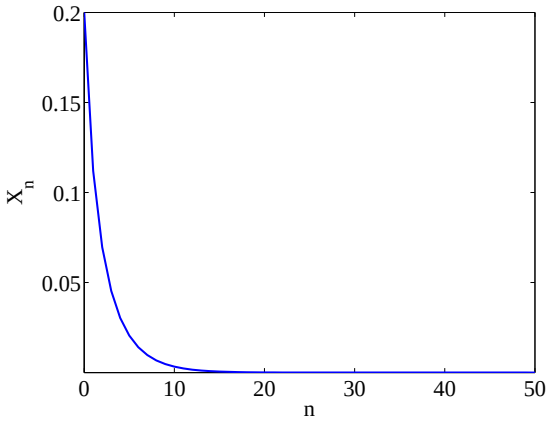


Figura 2.13: Resultado del mapa logístico para $r = 0{,}71$ y $x_0 = 0{,}2$.

Fuente:(el autor)

donde $x_0 = 0{,}2$ y r fue fijado en $0{,}7$. Para tasas de crecimiento de $1 < r < 3$ la población crece y eventualmente llega a un punto de equilibrio, donde se mantiene como se muestra en la Fig. 2.14.

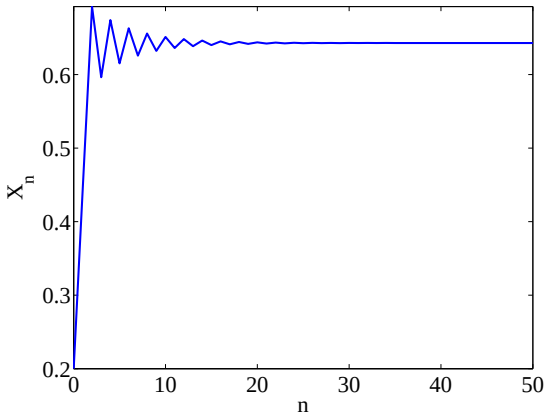


Figura 2.14: Resultado del mapa logístico para $r = 1{,}5$ y $x_0 = 0{,}2$.

Fuente:(el autor)

Para mayores tasas de crecimiento, por ejemplo, $r = 3{,}3$, la población vuelve a crecer pero luego de una generación vuelve a decrecer, manteniendo este estado de crecimiento y decrecimiento, como se muestra en la Fig. 2.15. Este tipo de oscilación, en la cual x_n se repite cada dos iteraciones, es llamado periodo de 2 ciclos.

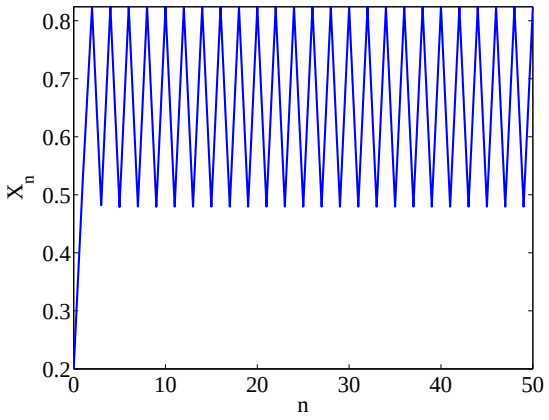


Figura 2.15: Resultado del mapa logístico para $r = 3{,}3$ y $x_0 = 0{,}2$.

Fuente:(el autor)

Con una tasa de crecimiento de $r = 3{,}5$, como se muestra en la Fig. 2.16, se puede observar que las oscilaciones continúan, pero los ciclos ahora se repiten cada 4 generaciones.

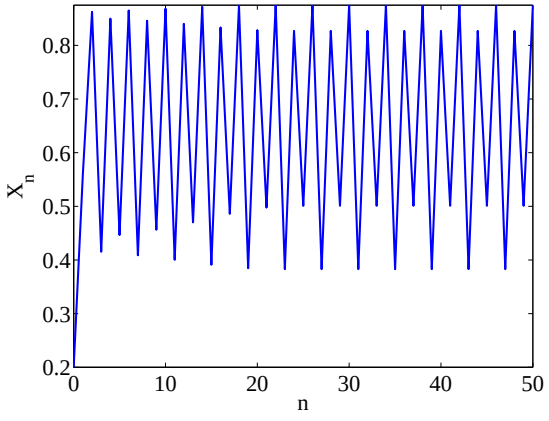


Figura 2.16: Resultado del mapa logístico para $r = 3{,}5$ y $x_0 = 0{,}2$.

Fuente:(el autor)

Los experimentos realizados demuestran que:

$r_1 = 3$	(2 periodos)
$r_2 = 3{,}449...$	4
$r_3 = 3{,}54409...$	8
$r_4 = 3{,}5644...$	16
$r_5 = 3{,}568759...$	32
$\vdots$	$\vdots$
$r_\infty = 3{,}569946...$	∞

Cuadro 2.1: Comparación del numero de periodos cuando se usan distintos valores de r.

Fuente:(el autor)

2.5.2. Caos y Ventanas Periódicas

Si le damos valores a r que se vayan acercando hacia r_∞, usando la simulación de la Fig. 2.12, se obtiene que, para algunos valores de r, la secuencia x_n nunca se queda en un punto de equilibrio ni tiene un comportamiento periódico. En la Fig. 2.17, donde $r = 3{,}9$, se puede observar que a largo plazo el comportamiento es aperiódico.

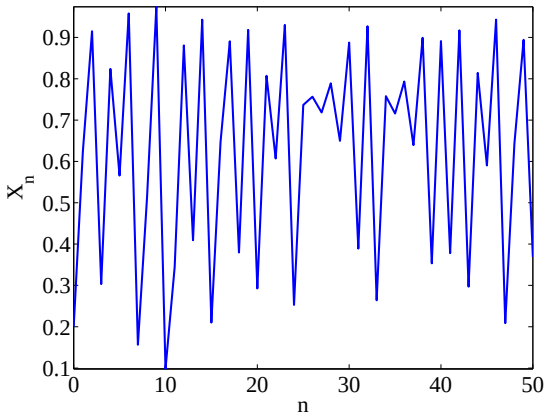


Figura 2.17: Resultado del mapa logístico para $r = 3{,}9$ y $x_0 = 0{,}2$.

Fuente:(el autor)

Su correspondiente diagrama de cobweb es el que se muestra en la siguiente figura

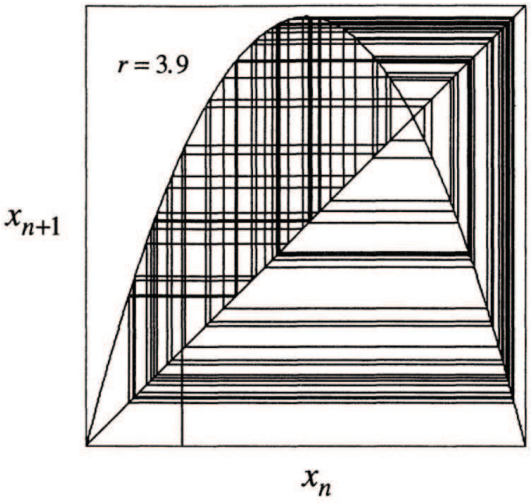


Figura 2.18: Diagrama de cobweb para $r = 3{,}9$ y $x_0 = 0{,}2$ [Strogatz, 2000].

Fuente:(el autor)

Donde se puede observar que el sistema se vuelve más caótico cada vez que r crece más.

2.6. Modulación

Muchas veces es necesarios transformar una señal antes de ser enviada o para que pueda ser transmitida por algún medio especifico. Como transformar la señal va depender mucho de su formato original y de el medio por el cual va ser transmitido.

Las señales deben ser modificadas, introduciéndose un cambio que pueda ser reconocido por el transmisor y el receptor y que representen la información enviada. Entre los tipos de conversiones tenemos [Forouzan, 2002]:

- Conversión digital a digital.
    - Unipolar.
    - Polar.
    - Bipolar.
    - Otros.
- Conversión de analógico a digital.
    - Modulación por amplitud de pulsos (PAM).
    - Modulación por codificación de pulsos (PCM).
    - Otros.
- Conversión de digital a analógico.
    - Modulación por desplazamiento de amplitud (ASK).
    - Modulación por desplazamiento de frecuencia (FSK).

- Modulación por desplazamiento de fase (PSK).
- Modulación por desplazamiento de fase (PSK).
- Modulación de amplitud en cuadratura (QAM).
- Otros

- Conversión de analógico a analógico.
  - Modulación en amplitud (AM).
  - Modulación en frecuencia (FM).
  - Modulación en fase (PM).
  - Otros

La modulación que se estudia en esta investigación es la modulación tipo delta (conversión analógico a digital) ya que es la que se propone para la implementación de una plataforma experimental que garantice la seguridad de las comunicaciones.

2.6.1. Modulador Tipo Delta con Variación Continua de Pendiente

La modulación tipo delta con variación continua de pendiente (CVDS) [Taylor, 1996] es un método simple y robusto de conversión A/D, para sistemas que requieren comunicación serial digital de señales analógicas. El modulador tipo delta consiste en un comparador y un integrador en la realimentación del lazo de control simple. La modulación delta (MD) esta limitada por la frecuencia analógica de entrada y por la amplitud procesada por el circuito. La frecuencia de reloj utilizada debe ser mínimo de 9600 Hz e ideal 64 kHz para aplicaciones de voz diseñado para aplicaciones típicas de

entradas analógicas de 1000 Hz.

En la Fig. 2.19, la salida digital, V_d, puede ser alta o baja en cualquier momento. Si V_d es alta, la salida del integrador tendrá un aumento gradual, caso contrario, si V_d es baja, la salida del integrador tendrá un decrecimiento gradual. Por lo que la señal de entrada analógica $m(t)$ es comparada con el voltaje de salida del integrador.

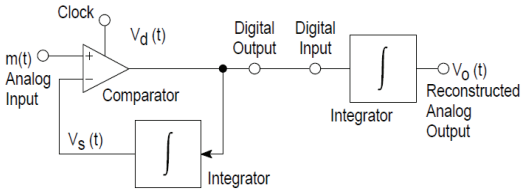


Figura 2.19: Modulador delta simple.

Fuente:([Taylor, 1996])

Para visualizar de manera más clara lo explicado, se tiene la Fig. 2.20 donde, se observa cuando $m(t)$ es mayor que v_s, v_d es alto y la salida del integrador tiene una pendiente positiva. Cuando v_s crece a un valor mayor que $m(t)$, v_d empieza a decrecer hasta que v_s de nuevo sea menor que $m(t)$ y el proceso se repite. La salida digital resultante, v_d, es el diferencial de la señal de entrada y también es la señal que sera transmitida al destino, donde sera integrada para poder obtener la señal analógica reconstruida. El tamaño del paso de tiempo, S, es el valor absoluto del cambio del voltaje de salida del integrador para un periodo de tiempo del reloj. Si la corriente de bits en serie del reloj es transmitida a un integrador similar que este ubicado en el destino, esta señal de

salida V_o, sería una copia del voltaje de salida del integrador del lazo cerrado, es decir, V_s.

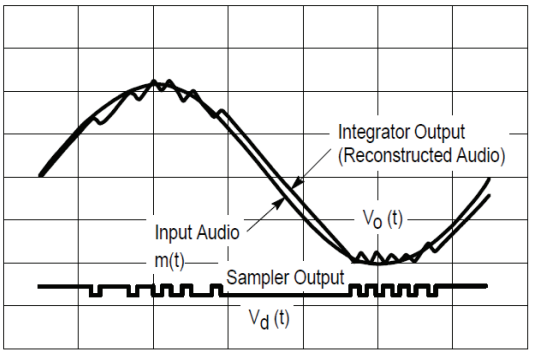


Figura 2.20: Ejemplo del comportamiento de las señales $m(t)$, V_d y V_o.

Fuente:([Taylor, 1996])

Lo que busca la CVDS es cambiar la pendiente de la señal de salida para que coincida con la pendiente de la señal de entrada, lo que permite que el ruido de cuantificación se mantenga minimizado. En la Fig. 2.21 se puede observar como el tamaño del paso de tiempo cambia con la amplitud de la señal de entrada.

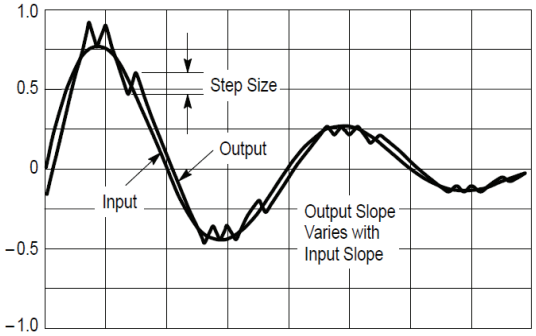


Figura 2.21: Ejemplo de Modulador/Demodulador (CVDS).

Fuente:([Taylor, 1996])

En las Figuras 2.22 y 2.23 se pueden observar el diagrama de bloques de un encoder y un decoder CVDS.

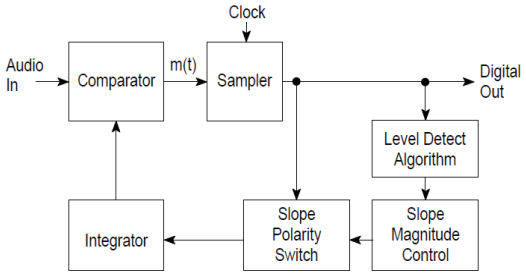


Figura 2.22: Diagrama de bloques de un encoder CVDS.

Fuente:([Taylor, 1996])

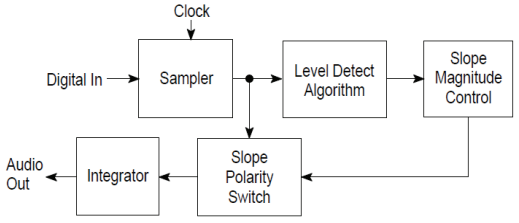


Figura 2.23: Diagrama de bloques de un decoder CVDS.

Fuente:([Taylor, 1996])

2.7. Descripción del Programa Arduino

Arduino [Arduino, 2015] es una tarjeta electrónica que se compone de un microcontrolador y puertos de comunicación. Estos puertos permiten transmitir y recibir datos. La programación del microcontrolador es sencilla gracias al lenguaje de programación Arduino que es muy similar al lenguaje C^{++}. Las razones por las cuales se usa Arduino como el hardware para probar los controladores son:

1. Asequible - Las tarjetas Arduino son relativamente baratas en comparación con otras plataformas de microcontroladores. La versión menos costosa del módulo Arduino puede ser ensamblada a mano, e incluso los módulos de Arduino premontados cuestan menos de 50$.

2. Multiplataforma - El software de Arduino se ejecuta en Windows, Macintosh OSX y sistemas operativos Linux.

3. De programación fácil - El entorno de programación de Arduino es fácil de usar para los principiantes y a su vez completo para que los usuarios avanzados saquen el máximo provecho.

4. De código abierto y hardware ampliable - Los diseñadores de circuitos con experiencia pueden hacer su propia versión del módulo, ampliándolo y mejorándolo.

La tarjeta usada en la investigación es el Arduino Uno R3, es la que se muestra en la Figura 2.24 la cual utiliza el microcontrolador ATmega328. El Arduino Uno utiliza el ATmega16U2 para el manejo de USB en lugar del 8U2 (o del FTDI encontrado en generaciones previas). Esto permite ratios de transferencia más rápidos y más memoria. No se necesitan drivers para Linux o Mac (el archivo inf para Windows es necesario y está incluido en el IDE de Arduino).

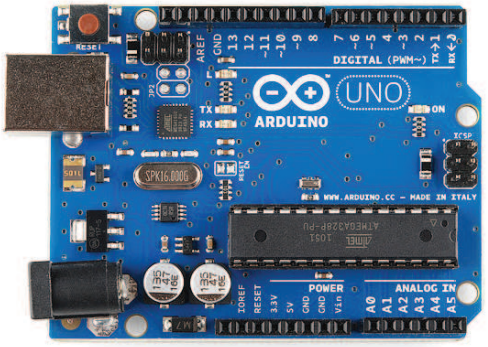


Figura 2.24: Tarjeta Arduino Uno.

Fuente:([Arduino, 2015])

La tarjeta Arduino Uno R3 incluso añade pins SDA y SCL cercanos al AREF. Es más, hay dos nuevos pines cerca del pin RESET. Uno es el IOREF, que permite a los shields adaptarse al voltaje brindado por la tarjeta. El otro pin no se encuentra conectado y está reservado para propósitos futuros. La tarjeta trabaja con todos los shields existentes y podrá adaptarse con los nuevos shields utilizando esos pines adicionales.

Las características de la tarjeta Arduino Uno se detallan a continuación [Arduino, 2015]:

1. Microcontrolador ATmega328.
2. Voltaje de entrada 7-12V.
3. 14 pines digitales de I/O (6 salidas PWM).

4. 6 entradas análogas.

5. 32k de memoria Flash.

6. Reloj de 16MHz de velocidad.

3. SIMULACIONES

Para las simulaciones se uso el programa Mathematica 6.0.3, DynPac. Lo primero que se realiza es definir un sistema para DynPac, usando el comando setmap, lo que le indica a DynPac que esto es un mapeo y no una ecuación diferencial.

setmap;

Luego se define las variable de estado, la función de mapeo y el nombre del sistema.

setstate[x];

setparm[r];

slopevec = r*r*(1-x); sysname= "Mapa Logístico";

Para revisar los datos del sistema se usa el comando sysreport.

sysreport

SYSTEM DEFINITION

System name: sysname = Mapa Logistico

State vector: statevec = $\{x\}$

State units: stateunits = $\{\}$

Slope vector: slopevec = $\{r * r * (1 - x)\}$

Parameter vector: parmvec = $\{r\}$

Parameter values: parmval = $\{r\}$

Parameter units vector: parmunits = $\{<\}$

Time unit: timeunit $=$

System Type: sysmode = mapping

Para empezar a ver el mapa logístico con distintos valores del parámetro r se usan los comandos,

parmval= 0.5;

viewmap[ ];

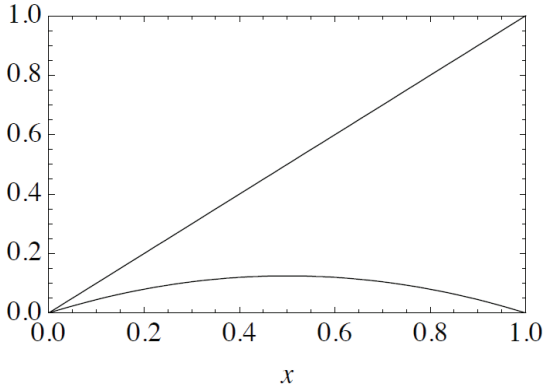


Figura 3.1: Resultado del mapa logístico para $r(1-x)x, \{r\} = \{0{,}5\}$.

Fuente:(el autor)

parmval= 2.0;

viewmap[ ];

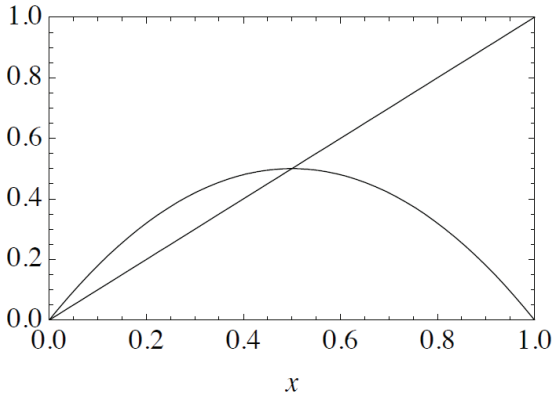


Figura 3.2: Resultado del mapa logístico para $r(1-x)x, \{r\} = \{2{,}0\}$.

Fuente:(el autor)

Las figuras 3.1 y 3.2, muestran que cuando se usa $r = 0{,}5$, existe un solo punto de equilibrio en el origen y cuando se usa $r = 2$ existen 2 puntos de equilibrio, uno $x = 0$ y otro en $x = 0{,}5$. Para diferentes valores de r, los puntos de equilibrio son:

findpolyfix[]

$\{\{0\}, \{\frac{-1+r}{r}\}\}$

Para determinar la estabilidad del sistema se usan los siguientes comandos:

eq0 = {0}; eq1 = {1-1/r};

parmval = {0.5};

classifymap[eq0]

strictly stable

Esto confirma lo observado anteriormente, que el mapa logístico cuando usa valores $1 < r < 3$ el sistema llega a un punto de equilibrio donde se mantiene.

parmval = 2.0;

classifymap[eq0]

unstable

classifymap[eq1]

strictly stable

Este resultado, indica que cuando en el mapa logístico $r = 2$, existen 2 puntos de equilibrio pero el único estable es el segundo.

parmval = 3.5;

classifymap[eq0]

unstable

classifymap[eq1]

unstable

Cuando $r > 3$ debido a la naturaleza del atractor, este sistema es inestable.

Para poder analizar el comportamiento de mapa logístico mediante el gráfico de cobweb, lo primero que se realiza es crear una secuencia de iteraciones mediante el comando iterate[init,initime,niter,ntoss,ncomp], donde init es el valor inicial de las ite-

raciones, initime es el tiempo de inicio y niter+ntoss es el numero total de iteraciones que se desea realizar.

En la siguiente prueba se usa el comanda para obtener 15 iteraciones, que arranquen desde $x = 0{,}8$ en $t = 0$, con un $r = 0{,}5$.

parmval = 0.5;

iterate[0.8, 0.0, 15, 0]

$\{\{0., 0.8\}, \{1., 0.08\}, \{2., 0.0368\}, \{3., 0.0177229\}, \{4., 0.00870439\}, \{5., 0.00431431\}, \{6., 0.00214785\}, \{7., 0.00107162\}, \{8., 0.000535235\}, \{9., 0.000267474\}, \{10., 0.000133701\}, \{11., 0.0000668417\}, \{12., 0.0000334186\}, \{13., 0.0000167088\}, \{14., 8.35424 * 10^{-6}\}, \{15., 4.17708 * 10^{-6}\}\}$

El resultado de este comando es una lista de pares, donde cada par esta conformado por {t, x}. Esta lista muestra que en cada iteración el sistema converge a 0. Para formar el gráfico de cobweb se usa el comando cobweb[init, initime, niter, ntoss, ncomp].

cobweb[0.8, 0.0, 15, 0]

En la Fig. 3.3 se puede observar como existe una convergencia hacia el punto de origen del sistema geométrico que se forma mediante el diagrama de cobweb.

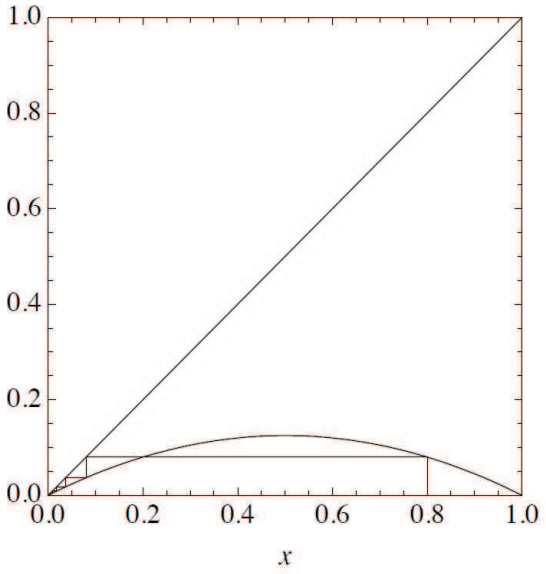


Figura 3.3: Diagrama de cobweb para $r = 0{,}5$

Fuente:(el autor)

A continuación se realizara el análisis cuando $r = 1{,}5$

parmval = 1.5;

Y con el comando eqstateval[ ], se podrá saber directamente el punto de equilibrio.

eqstateval[eq0]

{**0**}

eqstateval[eq1]

{**0.033333**}

Y realizando 30 iteraciones que empiecen desde $x = 0{,}5$ en $t = 0$,

iterate[0.05, 0.0, 30, 0]

{0., 0.05}, {1., 0.07125}, {2., 0.0992602}, {3., 0.134111}, {4., 0.174188}, {5., 0.21577

{6., 0.25382 {7., 0.284093}, {8., 0.305076}, {9., 0.318007}, {10., 0.325318¡, {11.,

0.329229}, {12., 0.331256}, {13., 0.332288}, {14., 0.332809}, {15., 0.333071}, {16.,

0.333202}, {17., 0.333268}, {18., 0.3333}, {19., 0.333317}, {20., 0.333325}, {21.,

0.333329}, {22., 0.333331}, {23., 0.333332}, {24., 0.333333}, {25., 0.333333}, {26.,

0.333333}, {27., 0.333333}, {28., 0.333333}, {29., 0.333333}, {30., 0.333333}}

Donde se vuelve a comprobar que los valores convergen al punto de equilibrio encontrado. Además realizando el diagrama de cobweb con 12 iteraciones se obtiene la Fig. 3.4 que muestra la misma convergencia.

cobweb[0.05, 0.0, 12, 0]

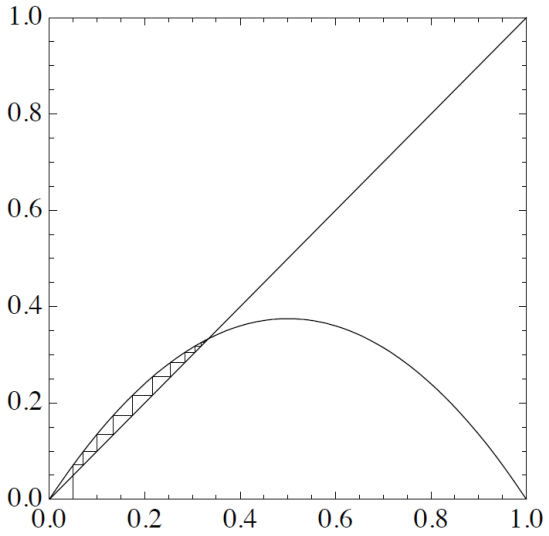


Figura 3.4: Diagrama de cobweb para $r = 1{,}5$.

Fuente:(el autor)

Analizando cuando $r = 2{,}5$ y realizando el mismo procedimiento se obtiene que el

punto de equilibrio esta en $x = 0{,}6$.

parmval = 2.5;

eqstateval[eq0]

{0}

eqstateval[eq1]

{0.6}

Las iteraciones quedan:

iterate[0.05, 0.0, 30, 0]

{{80., 0.05}, {1., 0.11875}, {2., 0.261621}, {3., 0.482939}, {4., 0.624272}, {5., 0.586391}, {6., 0.606341}, {7., 0.596729}, {8., 0.601609}, {9., 0.599189}, {10., 0.600404}, {11., 0.599798}, {12., 0.600101}, {13., 0.599949}, {14., 0.600025}, {15., 0.599987}, {16., 0.600006}, {17., 0.599997}, {18., 0.600002}, {19., 0.599999}, {20., 0.6}, {21., 0.6}, {22., 0.6}, {23., 0.6}, {24., 0.6}, {25., 0.6}, {26., 0.6}, {27., 0.6}, {28., 0.6}, {29., 0.6}, {30., 0.6}}

Y el gráfico de cobweb queda:

cobweb[0.0.5, 0.0, 30, 0]

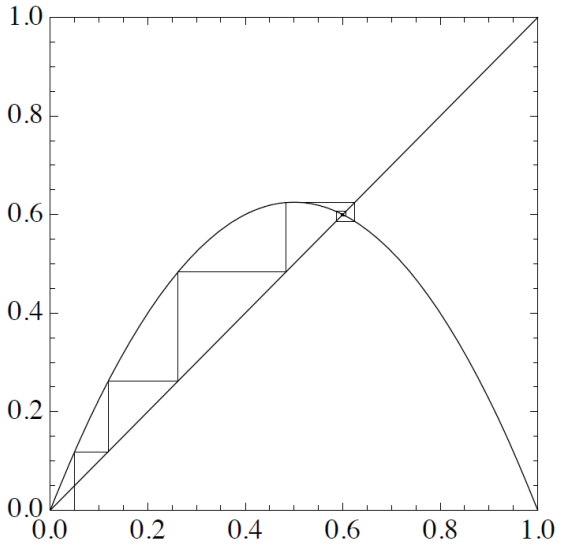


Figura 3.5: Diagrama de cobweb para $r = 2{,}5$.

Fuente:(el autor)

En las iteraciones y en el gráfico de cobweb se puede observar que este sistema converge a su punto de equilibrio $x = 0{,}6$ pero en las cercanías a este punto comienza a oscilar, lo cual se observo que pasa con los sistemas $1 < 1x < 3$.

Ahora se presentara el análisis cuando $r = 3{,}2$.

parmval = 3.2;

eqstate[eq0]

$\{0\}$

eqstate[equ1]

$\{0.6875\}$

mapval[eq0]

$\{0\}$

mapval[eq1]

{0.6875}

classifymap[eq0]

unstable

classifymap[eq1]

unstable

Para analizar el la estabilidad, se muestran los valores propios de las ecuaciones. El comando eigvalmap muestra los valores propios, que deben de ser menor a 1 en modulo para ser estables.

eigvalmap[eq0]

{3.2}

eigvalmap[eq1]

{-1.2}

Creando 20 iteraciones que empiecen en $x = 0{,}23$ y $t = 0$,

sol1=iterate[0.23, 0.0, 20, 0]

{{0., 0.23}, {1., 0.56672}, {2., 0.785755}, {3., 0.538701}, {4., 0.795207}, {5., 0.521129}, {6., 0.798571¡, {7., 0.514736}, {8., 0.799305}, {9., 0.513333}, {10., 0.799431¡, {11., 0.513091}, {12., 0.799452}, {13., 0.513052}, {14., 0.799455}, {15., 0.513046}, {16., 0.799455}, {17., 0.513045}, {18., 0.799455}, {19., 0.513045}, {20., 0.799455}}

Este resultado muestra que el sistema tiene oscilaciones y que x_n se repite cada dos iteraciones. Con el comando periodmap[] se puede confirmar lo indicado.

periodmap[sol1]

Solution contains a periodic orbit; period = 2

Mapeando este sistema usando un argumento de 2 para indicar el level de la composición de la función, se obtiene,

viewmap[2];

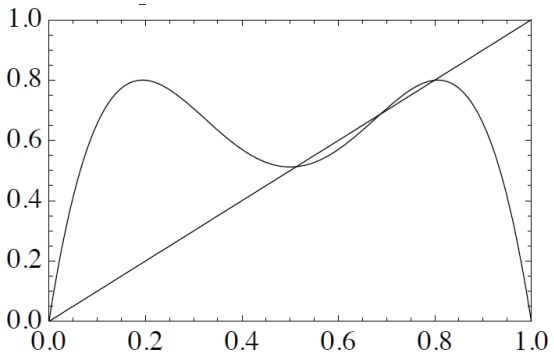


Figura 3.6: Viewmap con $r = 3{,}2$

Fuente:(el autor)

donde se pueden observar 4 puntos de equilibrio, donde 2 son los puntos de equilibrio del mapa logístico original y los otros dos indican los puntos de los periodos de 2 orbitas. Para saber exactamente cuales son estos puntos se usa el comando nfindpolyfix[2].

nfindpolyfix[2]

{{0.}, {0.513045}, {0.6875}, {0.799455}}

Si se analiza la estabilidad de los dos nuevos puntos se obtiene que ambos sons estric-

tamente estables.

classifymap[0.513045, 2]

strictly stable

classifymap[0.799455, 2]

strictly stable

En la Fig. 3.7 se muestra el gráfico de cobweb y se evidencia como el sistema oscila en los puntos $0{,}513045$ y $0{,}799455$.

cobweb[{0.23}, 100, 0];

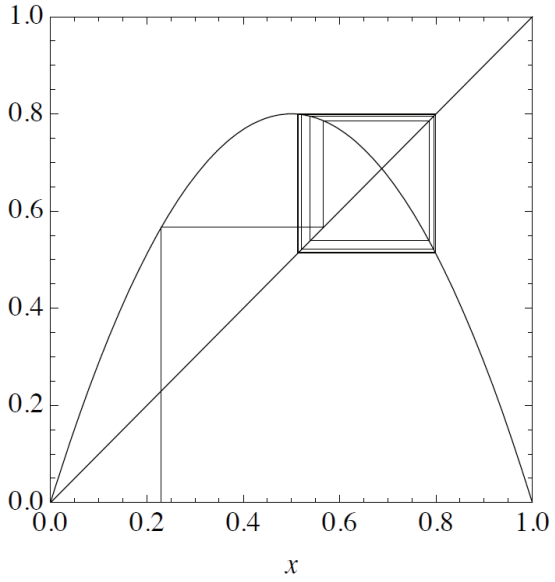


Figura 3.7: Diagrama de cobweb para $r = 3{,}2$.

Fuente:(el autor)

En la Fig. 3.8 se observa el diagrama de cobweb pero obteniendo solo la orbita.

cobweb[{0.23},100,100];

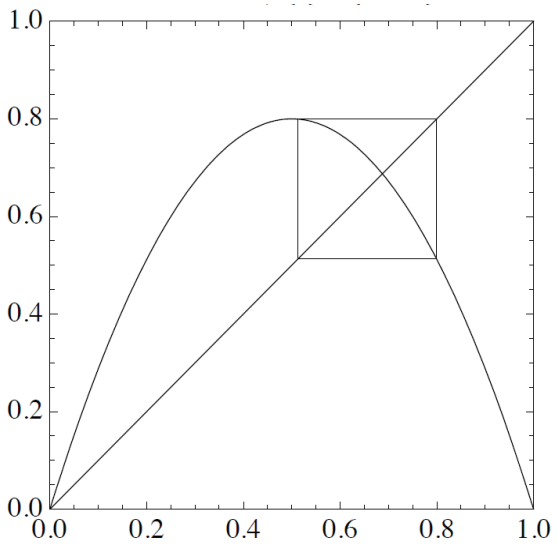


Figura 3.8: Diagrama de cobweb pero solo con la orbita cuando $r = 3{,}2$.

Fuente:(el autor)

Si se agranda r , por ejemplo $r = 3{,}5$.

parmval = 3.5;

sol2=[0.23, 0.0, 30, 0]

{{0., 0.23}, {1., 0.61985}, {2., 0.824726}, {3., 0.505936}, {4., 0.874877}, {5., 0.383136}, {6., 0.8272}, {7., 0.500291}, {8., 0.875}, {9., 0.382813}, {10., 0.826935}, {11., 0.500896}, {12., 0.874997}, {13., 0.38282}, {14., 0.826941}, {15., 0.500884}, {16., 0.874997}, {17., 0.38282}, {18., 0.826941}, {19., 0.500884}, {20., 0.874997}, {21., 0.38282}, {22., 0.826941}, {23., 0.500884}, {24., 0.874997}, {25., 0.38282}, {26., 0.826941}, {27., 0.500884}, {28., 0.874997}, {29., 0.38282}, {30., 0.826941}}

periodmap[sol2]

Solution contains a periodic orbit; period = 4

En las iteraciones y mediante el código periodmap se puede comprobar que este sistema tiene una orbita de 4 periodos.

Para observar los puntos fijos de las orbitas se agranda la imagen mediante el comando imsize y se gráfica el mapa logístico.

imsize = 320;

viewmap[4];

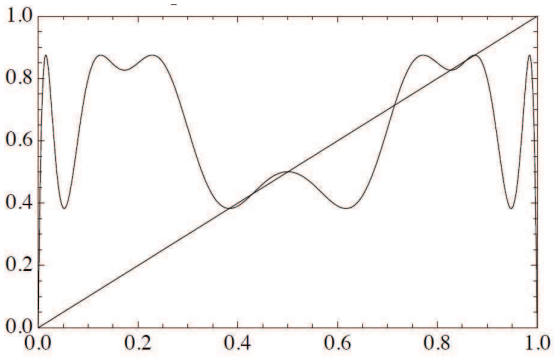


Figura 3.9: Resultado del mapa logístico con $r = 3{,}5$.

Fuente:(el autor)

Usando el comando nfindpolyfix[], en el mapa original, en el mapa segunda composición y de cuarta composición se encuentran los puntos fijos de las orbitas.

imsize = 250;

root1 = nfindpolyfix[]

{{0.},{0.714286}}

root2 = nfindpolyfix[2]

$\{\{0.\}, \{0.428571\}, \{0.714286\}, \{0.857143\}\}$

root3 = nfindpolyfix[4]

$\{\{0.\}, \{0{,}049385-0{,}0241573\iota\}, \{0{,}049385+0{,}0241573\iota\}, \{0{,}166354-0{,}0761994\iota\}, \{0{,}166354+$
$0{,}0761994\iota\}, \{0{,}38282\}, \{0{,}428571\},$
$\{0{,}500884\}, \{0{,}505703-0{,}177965\iota\}, \{0{,}505703+0{,}177965\iota\}, \{0{,}714286\}, \{0{,}826941\}, \{0{,}857143\},$
$0{,}00710473\iota\}, \{0{,}985737 + 0{,}00710473\iota\}\}$

Aquí se puede observar que el mapa original posee los dos puntos de equilibrios encontrados anteriormente, la segunda composición posee esos dos puntos de equilibrio más los puntos 0.428571 y 0.857143, los que constituyen periodo de dos orbitas. La cuarta composición tiene otros 4 adicionales, 0.38282, 0.500884, 0.826941 y 0.874997. Verificando la estabilidad de estos puntos:

classifymap[0]

unstable

classifymap[0.714286]

unstable

classifymap[0.428571, 2]

unstable

classifymap[0.857143, 2]

unstable

classifymap[0.38282, 4]

strictly stable

classifymap[0.500884, 4]

strictly stable

classifymap[0.826941, 4]

strictly stable

classifymap[0.874997, 4]

strictly stable

Produciendo 50 iteraciones que arranques desde el punto de inestabilidad 0.428571, se espera que al pasar el tiempo los valores se alejen de este puntos.

iterate[0.428571, 0.0, 50, 0]

{{0., 0.428571}, {1., 0.857143}, {2., 0.428572}, {3., 0.857143}, {4., 0.428571}, {5., 0.857143}, {6., 0.428572}, {7., 0.857143}, {8., 0.42857}, {9., 0.857142}, {10., 0.428573}, {11., 0.857144}, {12., 0.42857}, {13., 0.857142}, {14., 0.428573}, {15., 0.857144}, {16., 0.428569}, {17., 0.857142}, {18., 0.428575}, {19., 0.857144}, {20., 0.428567}, {21., 0.857141}, {22., 0.428576}, {23., 0.857145}, {24., 0.428565}, {25., 0.85714}, {26., 0.428579}, {27., 0.857147}, {28., 0.428562}, {29., 0.857138}, {30., 0.428584}, {31., 0.857149}, {32., 0.428556}, {33., 0.857135}, {34., 0.42859}, {35., 0.857152}, {36., 0.428548}, {37., 0.857131}, {38., 0.428601}, {39., 0.857158}, {40., 0.428534}, {41., 0.857124}, {42., 0.428618}, {43., 0.857166}, {44., 0.428513}, {45., 0.857114}, {46., 0.428644}, {47., 0.857179}, {48., 0.428481}, {49., 0.857097}, {50., 0.428685}}

Se puede observar que los valores se alejan del punto de equilibrio de forma lenta.

Analizando los valores propios se obtiene:

eigvalmap{0.428571, 2}

{-1.25001}

El valor propio encontrado es mayor que uno pero no por mucho por lo que se espera que tome un tiempo alejarse del punto.

Creando iteraciones que empiecen desde el punto 0.25 se obtiene:

iterate{0.25, 0.0, 50, 0}

{{0., 0.25}, {1., 0.65625}, {2., 0.789551}, {3., 0.581561}, {4., 0.851717}, {5., 0.442033}, {6., 0.863239}, {7., 0.4132}, {8., 0.84863}, {9., 0.449599¡, {10., 0.866109¡, {11., 0.405875}, {12., 0.843991}, {13., 0.460845}, {14., 0.869634}, {15., 0.396797}, {16., 0.837722}, {17., 0.475803}, {18., 0.872951}, {19., 0.388177}, {20., 0.831235}, {21., 0.490992}, {22., 0.874716}, {23., 0.383558}, {24., 0.827544}, {25., 0.499502}, {26., 0.874999}, {27., 0.382815}, {28., 0.826937}, {29., 0.500893}, {30., 0.874997}, {31., 0.38282}, {32., 0.826941}, {33., 0.500884}, {34., 0.874997}, {35., 0.38282}, {36., 0.826941}, {37., 0.500884}, {38., 0.874997}, {39., 0.38282}, {40., 0.826941}, {41., 0.500884}, {42., 0.874997}, {43., 0.38282}, {44., 0.826941}, {45., 0.500884}, {46., 0.874997}, {47., 0.38282}, {48., 0.826941}, {49., 0.500884}, {50., 0.874997}}

En la Fig. 3.10, se observa el diagrama de cobweb donde se muestra las orbitas luego de las primeras 50 iteraciones.

cobweb{0.25, 50, 0};

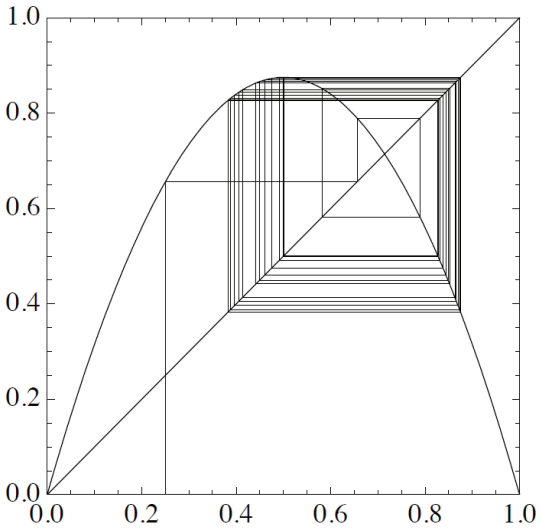


Figura 3.10: Diagrama de cobweb para $r = 3{,}5$.

Fuente:(el autor)

A continuación se analizara cuando $r > 3{,}499$.

parmval = 3.55;

sol8 = iterate[0.25, 0.0, 200, 0]

{{0., 0.25}, {1., 0.665625}, {2., 0.790118}, {3., 0.588703}, {4., 0.859568}, {5., 0.428523}, {6., 0.869363}, {7., 0.403176}, {8., 0.854219}, {9., 0.442077}, {10., 0.87559}, {11., 0.38671}, {12., 0.841937}, {13., 0.472431}, {14., 0.884802}, {15., 0.361843}, {16., 0.81974}, {17., 0.524571}, {18., 0.885357}, {19., 0.360325}, {20., 0.818243}, {21., 0.527961}, {22., 0.884725}, {23., 0.362054}, {24., 0.819946}, {25., 0.524102}, {26., 0.885438}, {27., 0.360104}, {28., 0.818023}, {29., 0.528458}, {30., 0.884625}, {31., 0.362326}, {32., 0.820212}, {33., 0.523497}, {34., 0.88554}, {35.,

0.359824}, {36., 0.817745}, {37., 0.529085}, {38., 0.884497}, {39., 0.362676}, {40., 0.820554}, {41., 0.52272}, {42., 0.885668}, {43., 0.359475}, {44., 0.817397}, {45., 0.52987}, {46., 0.884333}, {47., 0.363124}, {48., 0.82099}, {49., 0.521727}, {50., 0.885824}, {51., 0.359046}, {52., 0.816968}, {53., 0.530835}, {54., 0.884125}, {55., 0.363692}, {56., 0.821541}, {57., 0.52047}, {58., 0.886012}, {59., 0.35853}, {60., 0.816451}, {61., 0.531998}, {62., 0.883865}, {63., 0.364398}, {64., 0.822223}, {65., 0.518911}, {66., 0.88623}, {67., 0.357933}, {68., 0.81585}, {69., 0.533348}, {70., 0.883552}, {71., 0.365252}, {72., 0.823042}, {73., 0.517035}, {74., 0.88647}, {75., 0.357276}, {76., 0.815186}, {77., 0.534836}, {78., 0.883192}, {79., 0.366232}, {80., 0.823977}, {81., 0.514889}, {82., 0.886713}, {83., 0.356608}, {84., 0.814508}, {85., 0.536351}, {86., 0.882809}, {87., 0.367273}, {88., 0.824962}, {89., 0.512619}, {90., 0.886935}, {91., 0.355999}, {92., 0.813887}, {93., 0.537737}, {94., 0.882445}, {95., 0.368263}, {96., 0.825891}, {97., 0.510472}, {98., 0.887111}, {99., 0.355516}, {100., 0.813391}, {101., 0.53884}, {102., 0.882145}, {103., 0.369077}, {104., 0.82665}, {105., 0.508713}, {106., 0.88723}, {107., 0.355187}, {108., 0.813053}, {109., 0.539592}, {110., 0.881935}, {111., 0.369645}, {112., 0.827177}, {113., 0.507491}, {114., 0.887301}, {115., 0.354993}, {116., 0.812854}, {117., 0.540034}, {118., 0.88181}, {119., 0.369984}, {120., 0.82749}, {121., 0.506763}, {122., 0.887338}, {123., 0.354892}, {124., 0.81275}, {125., 0.540265}, {126., 0.881744}, {127., 0.370163}, {128., 0.827655}, {129., 0.506379}, {130., 0.887356}, {131., 0.354843}, {132., 0.812699}, {133., 0.540378}, {134., 0.881712}, {135., 0.37025}, {136., 0.827736}, {137., 0.506192}, {138., 0.887364}, {139., 0.35482}, {140., 0.812675}, {141., 0.540431}, {142., 0.881697}, {143., 0.370291}, {144., 0.827774}, {145., 0.506104}, {146., 0.887368}, {147., 0.354809}, {148., 0.812665}, {149., 0.540455},

{150., 0.88169}, {151., 0.37031}, {152., 0.827791}, {153., 0.506064}, {154., 0.887369}, {155., 0.354804}, {156., 0.81266}, {157., 0.540466}, {158., 0.881687}, {159., 0.370319}, {160., 0.827799}, {161., 0.506045}, {162., 0.88737}, {163., 0.354802}, {164., 0.812657}, {165., 0.540471}, {166., 0.881686}, {167., 0.370322}, {168., 0.827802}, {169., 0.506037}, {170., 0.887371}, {171., 0.354801}, {172., 0.812656}, {173., 0.540473}, {174., 0.881685}, {175., 0.370324}, {176., 0.827804}, {177., 0.506034}, {178., 0.887371}, {179., 0.354801}, {180., 0.812656}, {181., 0.540474}, {182., 0.881685}, {183., 0.370325}, {184., 0.827805}, {185., 0.506032}, {186., 0.887371}, {187., 0.354801}, {188., 0.812656}, {189., 0.540474}, {190., 0.881684}, {191., 0.370325}, {192., 0.827805}, {193., 0.506031}, {194., 0.887371}, {195., 0.354801}, {196., 0.812656}, {197., 0.540475}, {198., 0.881684}, {199., 0.370325}, {200., 0.827805}}

periodmap[sol8]

Solution contains a periodic orbit; period = 8

Los números encontrados muestran que la función genera una orbita periódica de 8 periodos, lo que se comprueba con el comando cobweb.

cobweb[0.25, 200, 0];

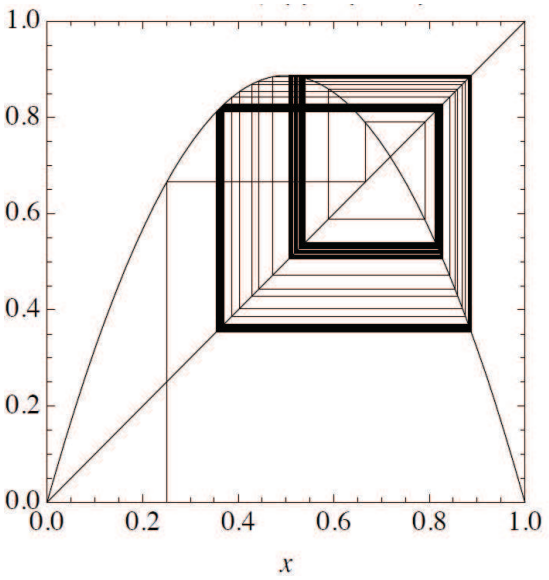


Figura 3.11: Diagrama de cobweb para $r = 3{,}55$.

Fuente:(el autor)

Ahora se trafican solo las orbitas.

cobweb[0.25, 100, 200];

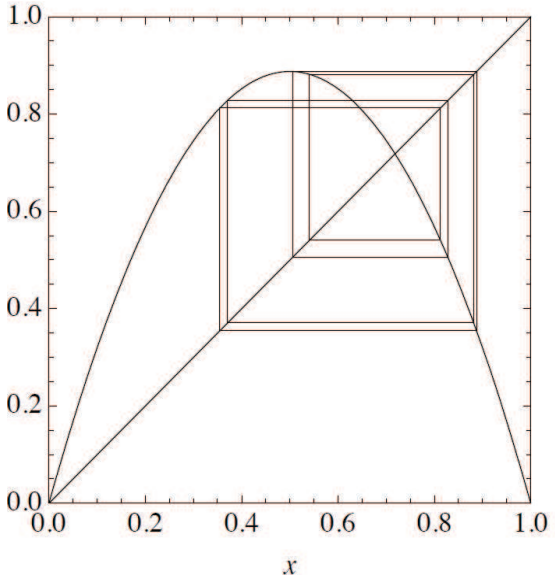


Figura 3.12: Diagrama de cobweb de las orbitas cuando $r = 3{,}55$.

Fuente:(el autor)

Para analizar las orbitas caóticas se da un valor de $r = 3{,}7$.

parmval = 3.7;

sol3 = iterate[0.23, 0, 100, 0]

{{0, 0.23}, {1, 0.65527}, {2, 0.835798}, {3, 0.507788}, {4, 0.924776}, {5, 0.257393}, {6, 0.707225}, {7, 0.766114}, {8, 0.662979}, {9, 0.82672}, {10, 0.530039}, {11, 0.921661}, {12, 0.267146}, {13, 0.724383}, {14, 0.738713}, {15, 0.714159}, {16, 0.755303}, {17, 0.683835}, {18, 0.799958}, {19, 0.592094}, {20, 0.893619}, {21, 0.351736}, {22, 0.843666}, {23, 0.488007}, {24, 0.924468}, {25, 0.25836}, {26, 0.708958}, {27, 0.763446}, {28, 0.668206}, {29, 0.820315}, {30, 0.545374}, {31, 0.917383}, {32, 0.28043}, {33, 0.746619}, {34, 0.699963}, {35, 0.777055}, {36, 0.64099}, {37, 0.851451}, {38, 0.467984}, {39, 0.921207}, {40, 0.268562}, {41,

0.726815}, {42, 0.734653}, {43, 0.72127}, {44, 0.743846}, {45, 0.704995}, {46, 0.769515}, {47, 0.656238}, {48, 0.834682}, {49, 0.510555}, {50, 0.924588}, {51, 0.257983}, {52, 0.708283}, {53, 0.764487}, {54, 0.666172}, {55, 0.822831}, {56, 0.539387}, {57, 0.91926}, {58, 0.274618}, {59, 0.73705}, {60, 0.717086}, {61, 0.750632}, {62, 0.69258}, {63, 0.787779}, {64, 0.618579}, {65, 0.872974}, {66, 0.410293}, {67, 0.895225}, {68, 0.347049}, {69, 0.838443}, {70, 0.501189}, {71, 0.924995}, {72, 0.256704}, {73, 0.705986}, {74, 0.768008}, {75, 0.659235}, {76, 0.831183}, {77, 0.519175}, {78, 0.92364}, {79, 0.260959}, {80, 0.71358}, {81, 0.756219}, {82, 0.682102}, {83, 0.802304}, {84, 0.586865}, {85, 0.897082}, {86, 0.341606}, {87, 0.832172}, {88, 0.516748}, {89, 0.923962}, {90, 0.259947}, {91, 0.711787}, {92, 0.759042}, {93, 0.67672}, {94, 0.809449}, {95, 0.570692}, {96, 0.90651}, {97, 0.313575}, {98, 0.796409}, {99, 0.599925}, {100, 0.888056}}

Se puede observar que no existen valores que se repiten y para garantizar eso se revisa si existen periodos.

periodmap[sol3]

Solution does not contain a periodic orbit.

Se revisan los puntos deequilibrio.

nfindpolyfix[]

{{0.}, {0.72973}}

nfindpolyfix[2]

$\{\{0.\}, \{0.390022\}, \{0.72973\}, \{0.880248\}\}$

nfindpolyfix[4]

$\{\{0.\}, \{0{,}0447134-0{,}015327\iota\}, \{0{,}0447134+0{,}015327\iota\}, \{0{,}158911-0{,}0516384\iota\}, \{0{,}158911+$

$0{,}0516384\iota\}, \{0{,}321626\}, \{0{,}390022\}, 80{,}504402-0{,}130338\iota\}, \{0{,}504402+0{,}130338\iota\}, \{0{,}575652\}$

$0{,}00424622\iota\}, \{0{,}987784 + 0{,}00424622\iota\}\}$

A pesar de que se encontraron puntos de equilibrio se espera que todos sean inestables.

classifymap[{0}]

unstable

classifymap[{0.72973}]

unstable

classifymap[{0.390022}, 2]

unstable

classifymap[{0.321626}, 4]

unstable

cobweb[{0.23}, 200, 0];

Haciendo el diagrama de cobweb se obtiene:

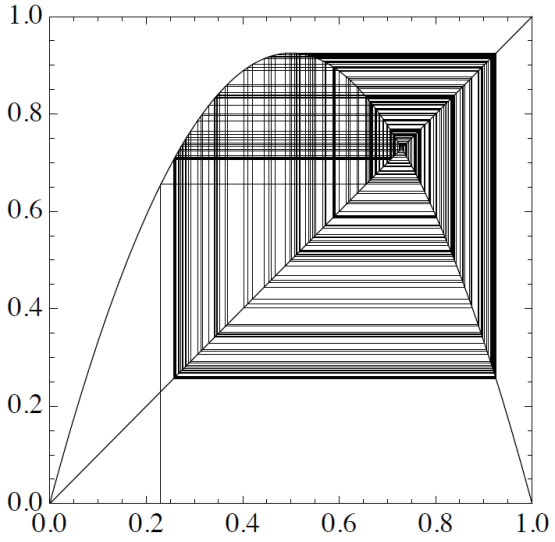


Figura 3.13: Diagrama de cobweb para $r = 3{,}7$.

Fuente:(el autor)

Los resultados se pueden resumir en que cuando el parámetro de crecimiento r es menor que 3, existe un único punto de equilibrio estable, y en general la iteración convergerá a ese punto. Una vez que r excede a 3, no hay puntos de equilibrio estables, pero en el intervalo $3 < r < 3{,}569946...$, siempre hay una solución periódica estable. Encontramos una secuencia de dobles bifurcaciones del período, de modo que las órbitas estables son cada vez más largas. Finalmente cuando r excede $3,569946...$, ya no hay soluciones periódicas estables. Una manera informal de describirlo es decir que parecen haber alcanzado una solución oscilatoria con un período infinito, y eso suena como caos. La siguiente tabla muestra los resultados:

Sistema $\{r * r * (1 - x)\}$	
Rango de parámetros	**Tipo de atractor**
$0 < r < 1$	Equilibrio a $r = 0$.
$1 < r < 3$	Equilibrio a $1 - 1/r$
$3 < r < 3{,}449...$	Orbita de 2 periodos
$3{,}449... < r < 3{,}54409...$	Orbita de 4 periodos
$3{,}54409... < r < 3{,}5644...$	Orbita de 8 periodos
$3{,}5644... < r < 3{,}568759...$	Orbita de 16 periodos
.....	.....
$3{,}569946... < r$	?

Cuadro 3.1: Resultados de las simulaciones.

Fuente:(el autor)

4. PLATAFORMA EXPERIMENTAL USANDO MAPAS LOGÍSTICOS

El objetivo de esta sección es presentar un esquema experimental del uso de señales caóticas para garantizar la seguridad de las comunicaciones usando un microcontrolador [Zapateiro De la Hoz et al., 2015]. Para esto se va usar el mapa logístico como medio de seguridad, para encriptar la información, además que se genera una llave de seguridad que también estará encriptada. El esquema propuesto es el que se muestra en la Fig. 4.1 y consiste en los siguientes bloques:

- Transmisor (Microcontrolador).El microcontrolador debe obtener el mensaje $m(t)$ que se desea transmitir por uno de sus puertos de entrada analógicas. El microcontrolador toma muestras del mensaje de en entrada, $m(t)$ y lo convierte en una señal muestreada (digital), llamada $m(k)$, $k = nT$, T es el tiempo de muestreo y $n = 0, 1, 2, ...$ Esta señal es encriptada usando el mapa logístico y un modulador delta simple. Luego de esto, una señal tipo llave, $s(k)$, es generada para desencriptar el mensaje en el receptor. Esta señal $s(k)$ es también es encriptada mediante un segundo mapa logístico. Esto quiere decir que el microcontrolador, genera 3 señales de salida:

  - El mensaje encriptado, $m_e(k)$
  - La señal tipo llave de encriptación $s_e(k)$
  - Una señal tipo llave de desencriptación $s_1(k)$

- Canales. Los 3 canales cableados para enviar el mensaje y las señales tipo llave.

- Receptor (Microcontrolador). Este elemento se encarga de recibir las señales $m_e(k)$, $s_e(k)$ y $s_1(k)$ para desencriptar la señal delta-modulada antes de ser convertida en una señal analógica. La salida es una señal digital, $m_d(k)$.

- Demodulador Delta. El demodulador delta consiste de un integrador, un filtro y algunos amplificadores para recuperar el mensaje original. La salida es una señal, $m_r(t)$ que es una aproximación de la señal original $m(t)$.

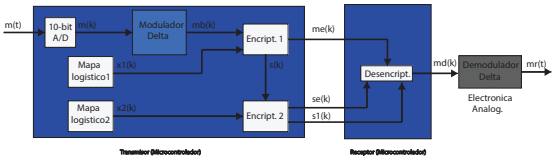


Figura 4.1: Esquema de la plataforma experimental.

Fuente:(el autor)

Para la descripción de la implementación del sistema de seguridad se usa como ejemplo el microprocesador Arduino como transmisor y receptor. La comunicación empieza cuando un mensaje $m(t)$ es generado y es enviado a una de las entradas analógicas del microcontrolador. Las entradas analógicas del microcontrolador Arduino solo aceptan señales unipolares en el rango de voltaje 0V a 5V, por lo que esta señal es convertida de analógica a digital mediante un convertidor ADC de 10 bits (elemento que el microprocesador tiene internamente) a una tasa de 10000 muestras por segundo. A la salida del ADC se va obtener una señal entre 0 y 1023, esta será la señal $m(k)$.

La modulación se la realiza mediante el modelo del modulador Delta, la cual se la puede ver como una conversión de un ADC de 1 bit, por lo que genera un bit de salida

por muestra de entrada. A la salida del modulador tipo delta se obtiene una señal $m_b(k)$.

Para la encriptación de la señal $m_b(k)$ se usan dos mapas logísticos que generan dos valores, $x_1(k)$ y $x_2(k)$. Los dos mapas logísticos tienen distintas condiciones iniciales, es decir $x_1(0) \neq x_2(0)$, pero ambos están configurados con el parámetro $r = 3{,}9$ para que tengan un comportamiento caótico. Primero, el mensaje es codificado con un valor VERDADERO o FALSO que es asignado dependiendo el valor de $x_1(k)$ generado por el primer mapa logístico. La rutina es la siguiente:

Si $x_1(k) > 0.5$ entonces

$me(k) = mb(k)$

$s(k) =$ verdad

caso contrario

$me(k) =!mb(k)$ (el símbolo ! es una negación booleana).

$s(k) =$ falso

fin

donde $m_e(k)$ es el mensaje encriptado y $s(k)$ es la llave generada que sirve para recuperar la señal $m_e(k)$.

Para aumentar la seguridad del sistema, la llave $s(k)$ también es encriptada siguiendo el mismo esquema. Esto se hace asignando un valor VERDADERO o FALSO dependiendo del valor del segundo mapa logístico, cuya rutina puede ser:

Si $x_2(k) < 0.1$ entonces

$s_1(k) =!s(k)$

$s_2(k) =$ verdad

caso contrario

$s_1(k) = s(k)$

$s_2(k) =$ falso

fin

donde $s_1(k)$ y $s_2(k)$ son señales auxiliares que son usadas para encriptar y desencriptar la llave de seguridad $s(k)$. Esta llave es encriptada mediante la función:

$$s_e(k) = (!s_1(k) \text{ AND } s_2(k)) \text{ OR } (s_1(k) \text{ AND } !s_2(k))$$

Las señales $s_e(k)$, $m_e(k)$ y $s_1(k)$ son enviadas al receptor mediante las salidas digitales del microprocesador.En el receptor se desencripta la señal $s_e(k)$ aplicando la técnica del mapa de Karnaugh [Karnaugh, 1953] , para así obtener $s_2(k)$, quedando una función:

$$s_2(k) = (!s_1(k) \text{ AND } s_e(k)) \text{ OR } (s_1(k) \text{ AND } !s_e(k))$$

Obteniendo $s_2(k)$ la señal $s(k)$ mediante la siguiente rutina:

Si $s_2(k)$ = VERDAD entonces

$s(k) =!s_1(k)$

caso contrario

$s(k) = s_1(k)$

fin

Finalmente la señal $m_e(k)$ es desencriptada analizando el valor de $s(k)$:

Si $s(k)$ = VERDAD entonces

escribe en el puerto de salida digital: $m_d(k) = m_e(k)$

caso contrario

escribe en el puerto de salida digital: $m_d(k) =!m_e(k)$

fin

donde $m_d(k)$ es la señal desencriptada. Esta señal es enviada al demodulador tipo delta (aplicando electrónica analógica) y este entrega una señal $m(t)$. Con lo cual se recupera la señal enviada de una manera segura usando el mapa logístico para garantizar la seguridad en la comunicación.

5. CONCLUSIONES, RECOMENDACIONES Y FUTUROS TRABAJOS

5.1. Conclusiones

Como se indica en el informe anual de seguridad 2016 en [CISCO, 2016], los atacantes y los responsables de la seguridad están desarrollando tecnologías y tácticas cada vez más sofisticadas. Por su parte, los atacantes están creando infraestructuras back-end sólidas para el lanzamiento y soporte de sus campañas. Los ciberdelincuentes están perfeccionando sus técnicas para obtener dinero de sus víctimas y para evitar ser detectados mientras continúan robando davtos y propiedad intelectual. Por lo que es importante garantizar la seguridad de todos los tipos de comunicaciones usando nuevas técnicas de encriptación.

Es este trabajo se presentó la implementación de una plataforma experimental que garantice la seguridad de las comunicaciones usando sistemas caóticos. Es importante tener claro los conceptos básicos de sistemas y señales caóticas para poder implementar este sistema por lo que en este trabajo se muestran las bases conceptuales de un sistema, indicando dos de sus tipos de configuración, más usados. Además se muestra el estudio de sistemas caóticos incluyendo sus tipos y sus propiedades, destacando su comportamiento aleatorio sin un modelo establecido y sensible a distintas condiciones iniciales, por lo cual puede ser usado para encriptar mensajes.

El uso de los mapas logísticos como medio para encriptar los mensajes permite garan-

tizar la seguridad de las comunicaciones, además que el modelo digital ayuda a poder realizar una sencilla implementación en un microcontrolador para comprobar su comportamiento y verificar que cumpla la seguridad deseada.

En las simulaciones realizadas, se comprueba como un sistema caótico puede cambiar su comportamiento dependiendo de las condiciones iniciales que tenga. Este aspecto es el que le da una gran importancia para poder usado en la encriptación de las comunicaciones.

Se presentaron varios tipos de modulaciones, destacando entre ellas a la modulación tipo delta con variación continua de pendiente la cual es usada en conjunto con el microcontrolador Arduino Uno para poder montar un sistema que garantice la seguridad de las comunicaciones usando señales caóticas.

Este sencillo esquema de la aplicación del mapa logístico y del modulador/demosulador Delta para garantizar la seguridad de las comunicaciones puede ser montado y probado en una institución académica como equipo de prueba, de una forma económica y fácil.

5.2. Recomendaciones

A pesar de que el tema ”seguridad de las comunicaciones.es muy importante a nivel mundial, muchas empresas, instituciones, personas, etc. siguen usando los mismos métodos de seguridad, habituales y de fácil acceso para los atacantes. Existe muy poco conocimiento acerca de nuevas tecnologías de seguridad de la comunicación además de que no existe un incentivo a la investigación de nuevas técnicas de encriptación de señales. Principalmente, muchos conocen lo que es seguridad de las comunicaciones, por los equipos que compran en el mercado pero no saben los aspectos tan importantes como los fundamentos técnicos de los mismos, las características de los soportes de transmisión que utiliza y las realidades de la tecnología.

Ya que la plataforma presentada en esta investigación es económica, de fácil desarrollo, con una alta fiabilidad y ya que usa técnicas novedosas como es el uso de sistemas caóticos, puede ser montada en un laboratorio de la universidad para enseñanza s estudiantes de nuevos métodos de seguridad. Dentro de esta enseñanza pueden existir varias etapas:

- Aprendizaje de técnicas de seguridad.
- Decodificación y codificación.
- Desarrollo de plataformas.
- Programación de microcontroladores.
- Pruebas de laboratorio .

Además que estas técnicas pueden ser mejoradas y se enseña al estudiante a desarrollar sus propios equipos y no solo a usar lo que se compran en el mercado.

5.3. Futuros trabajos

En los futuros trabajos, están:

- Usar distintos microcontroladores y ver sus rendimientos.
- Probar nuevas técnicas de decodificación y codificación.
- Explorar las señales caóticas y sus propiedades para hacer más robusto el sistema.
- Realizar publicaciones en revistas indexadas, de los resultados obtenidos en cada una de las investigaciones.

Referencias

[Alvarez, 2006] Alvarez, G.; Shujun, L. (2006). Some basic cryptographic requirements for chaos-based cryptosystems. *International Journal of Bifurcation and Chaos*, 16(2129-2152).

[Andrievsky, 2002] Andrievsky, B. (2002). Adaptative synchronization methods for signal transmission on cahotic carriers. *Mathematics and Computers in Simulation*, 58:285–293.

[Arduino, 2015] Arduino (2015).

[Argyris and Mirasso, 2005] Argyris, A.; Syvridis D.; Larger, L. A. V. C. P. F. I. G. J. and Mirasso, R.; Pesquera, L. S. A. (2005). Chaos-based communications at high bit rates using comercial fibre-optic links. *Nature*, 438:343–346.

[CISCO, 2016] CISCO (2016). Informe anual de seguridad. Technical report.

[Cuomo, 1992] Cuomo, K.; Oppenheim, A. (1992). Synchronized chaotic circuits and systems for communications. 575, MIT Research Laboratory.

[Cuomo, 1993a] Cuomo, K.; Oppenheim, A. (1993a). Circuit implementation of synchroized chaos, with applications to communications. *Phys. Rev. Lett.*, 71:65.

[Cuomo, 1993b] Cuomo, K.; Oppenheim, A. (1993b). Synchronized of lorenz-based chaotic circuits, whit applications to communications. *IEEE Trans. Circuits and Systems*.

[Forouzan, 2002] Forouzan, B. (2002). *Transmision de datos y redes de comunicaciones.* McGraw-Hill Interamericana de Espana.

[Fradkov, 2000] Fradkov, A.; Nijmeijer, H. M. A. (2000). Adaptative observer-based synchronization for communication. *International Journal of Bifurcation and Chaos*, 10:2807–2813.

[Karnaugh, 1953] Karnaugh, M. (1953). The map method for synthesis of combinational logic circuits. *Transactions of the American Institute of Electrical Engineers, PArt I: Communications and Electronics*, 72:593–599.

[Kocarev, 1992] Kocarev, L.; Halle, K. E. K. P. U. (1992). Experimental demostration of secure communications via chaotic synchronization. *International Journal of Bifurcation and Chaos*, 2:709–713.

[Li, 2004] Li, Z.; Xu, D. (2004). A secure communication scheme using projective chaos synchronization. *Chaos, Solitons & Fractals*, 22:477–481.

[Moez, 2003] Moez, F. (2003). An adaptive chaos synchronization scheme applied to secure communication. *Chaos, Solitons & Fractals*, 18:141–148.

[Moriello, 2015] Moriello, S. (2015). Sistemas complejos, caos y vida artifical. *RED-cientifica*.

[Murillo, 2014] Murillo, M.; Abundiz, F. (2014). A novel symmetric text encryption algorithm based on logistic map. *Proceedings of the 2014 International Conference on Communications, Signal Processing and Computers*, pages 49–53.

[Pande, 2010] Pande, A.; Zambreno, J. (2010). Design and hardware implementation of a chaotic encryption scheme for real-time embedded systems. *IEEE Signal Processing and Communications (SPCOM)*.

[Parlitz, 1992] Parlitz, U.; Chua, L. K. L. H. L. S. A. (1992). Transmission of digital signals by chaotic synchronization. *International Journal of Bifurcation and Chaos*, 2:973–977.

[Pecora, 1989] Pecora, L.; Carroll, T. (1989). Synchronization in chaotic systems. *Physical Review Letters*, 64:821–825.

[Strogatz, 2000] Strogatz, S. (2000). *Nonlinear dynamics and chaos with applications to physics, bilogy, chemistry, and engineering.* Westview Press, 1 edition.

[Taylor, 1996] Taylor, D. (1996). Design of continuously variable slope delta modulation communication systems. An1544, Motorola.

[Verhulst, 1845] Verhulst, P. (1845). Recherches mathematiques sur la loi d acroissement de la population. *Memoires de l AcadEmie Royale des Sciences, des Lettres et des Beaux Arts de Belgique*, 18:1–38.

[Volos, 2013] Volos, C. (2013). Chaotic random bit generator realized with a microcontroller. *Journal of Computations & Modelling*, 3:115–136.

[Yang, 1996] Yang, T.; Chua, L. (1996). Secure communications via chaotic parameter modulation. *IEEE Transactions on Circuits and Systems*, 43:817–819.

[Zapateiro, 2014] Zapateiro, M.; Acho, L. V. Y. (2014). A modified chua chaotic oscillator and its application to secure communications. *Applied Mathematics and Computation*, 247:712–722.

[Zapateiro De la Hoz et al., 2015] Zapateiro De la Hoz, M., Acho, L., and Vidal, Y. (2015). An experimental realization of a chaos-based secure communication using arduino microcontrollers. *The Scientific World Journal*, 2015.

GLOSARIO

A/D — Analógico-Digital.

AM — Modulación en amplitud.

ASK — Modulación por desplazamiento en amplitud).

CVDS — Modulación tipo delta con variación continua de pendiente.

FM — Modulación en frecuencia.

FSK — Modulación por desplazamiento en frecuencia.

IDE — Ambiente de desarrollo integrado.

MD — Modulación delta.

PAM — Modulación por amplitud de pulsos.

PCM — Modulación por codificación de pulsos.

PCK — Modulación por desplazamiento de fase.

PM — Modulación en fase.

QAM — Modulación en cuadratura.

DECLARACIÓN Y AUTORIZACIÓN

Yo, **Tutivén Gálvez, Christian Javier**, con C.C: # **0914742267** autor/a del trabajo de titulación: "**Estudio de la sincronización de sistemas caóticos para garantizar la seguridad de las comunicaciones"** previo a la obtención del título de **Magíster en Telecomunicaciones** en la Universidad Católica de Santiago de Guayaquil.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, 18 de mayo de 2018

f. _____

Nombre: **Tutivén Gálvez, Christian Javier**

C.C: **0914742267**

REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA			
FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN			
TÍTULO Y SUBTÍTULO:	Estudio de la sincronización de sistemas caóticos para garantizar la seguridad de las comunicaciones		
AUTOR(ES)	Tutivén Gálvez, Christian Javier		
REVISOR(ES)/TUTOR	MSc. Orlando Philco Asqui; MSc. Luis Córdova Rivadeneira / MSc. Manuel Romero Paz		
INSTITUCIÓN:	Universidad Católica de Santiago de Guayaquil		
FACULTAD:	Sistema de Posgrado		
PROGRAMA:	Maestría en Telecomunicaciones		
TITULO OBTENIDO:	Magister en Telecomunicaciones		
FECHA DE PUBLICACIÓN:	**Guayaquil, 18** de **mayo** de **2018**	**No. DE PÁGINAS:**	**82**
ÁREAS TEMÁTICAS:	Sistema, Sistemas Caóticos, Ecuación de Lorenz, Simetría, No-Linealidad, Modulación, Programa Arduino		
PALABRAS CLAVES/ KEYWORDS:	Sistema, Sistemas Caóticos, Simetría, No-Linealidad, Modulación, Arduino		

RESUMEN/ABSTRACT: Todos los días se realizan millones de comunicaciones y la información se ha convertido en un activo de gran valor para las personas y las instituciones. Por lo que garantizar la seguridad de la información que se transmite es de vital importancia para así no sufrir pérdidas ni robos de la misma que puedan causar daños económicos y de confianza. Muchos métodos se han investigado y desarrollado a lo largo de los años pero la mayoría usan técnicas que usan modelos que son fácil de predecir y de decodificar. Por lo que el uso de técnicas que usen modelos matemáticos con comportamientos caóticos ha ganado mucho interés. En esta tesis se presenta un estudio de las señales caóticas y su uso para garantizar la seguridad de las comunicaciones, además que se realiza un esquema de fácil implementación en un microprocesador para poder realizar pruebas y prácticas de laboratorio. En el capítulo 1, se hace referencia a los aspectos generales de la investigación. En el capítulo 2 se presenta un estudio de las señales caóticas, del sistema de modulación aplicado y del microprocesador usado en el esquema experimental. En el capítulo 3, se realizan distintas simulaciones, usando el programa Mathematica para comprobar el comportamiento de las señales caóticas y del mapa logístico. En el capítulo 4 se presenta, un esquema propuesto de fácil implementación del mapa logístico para garantizar la seguridad de la comunicación. Finalmente, en el capítulo 5 se presentan conclusiones y recomendaciones de la investigación realizada.

ADJUNTO PDF:	☒ SI	☐ NO
CONTACTO CON AUTOR/ES:	**Teléfono:** +593-991904393	**E-mail:** Christian_tutiven@hotmail.com
CONTACTO CON LA INSTITUCIÓN (C00RDINADOR	**Nombre:** Romero Paz Manuel de Jesús	
	Teléfono: +593-994606932	

DEL PROCESO UTE)::	**E-mail:** manuel.romero@cu.ucsg.edu.ec
SECCIÓN PARA USO DE BIBLIOTECA	
N°. DE REGISTRO (en base a datos):	
N°. DE CLASIFICACIÓN:	
DIRECCIÓN URL (tesis en la web):	