

Una propuesta criptográfica basada en atractores caóticos

Una propuesta criptográfica basada en atractores caóticos

Deicy Alvarado Nieto

Isabel Amaya Barrera

César Suárez Parra



Resumen

Este libro surgió de un ejercicio académico que contó con la participación de estudiantes y profesores del programa de Ingeniería de Sistemas de la Universidad Distrital Francisco José de Caldas. Tiene como principal objetivo aplicar los principios de los sistemas dinámicos caóticos en el contexto de la seguridad informática. Para ello, aborda temáticas como atractores caóticos, teoría de cuaterniones y autómatas celulares, lo cual facilita el manejo de información de gran tamaño y la generación de propuestas criptográficas y esteganográficas con este enfoque. Los modelos presentados en este libro fueron aplicados a diversos archivos de gran tamaño, tales como imágenes, audios, textos, videos y archivos multimedia en general. Estos modelos fueron sometidos a evaluaciones de seguridad y desempeño, logrando indicadores acordes con los reportados en la literatura científica actual. Esto evidencia que tales propuestas podrían, eventualmente, ser implementadas en entornos reales. Dado el vertiginoso avance de las tecnologías, la seguridad de la información se convierte en un problema social de gran impacto a nivel mundial. Este hecho motiva a la comunidad académica a trabajar de manera interdisciplinaria en el liderazgo de propuestas robustas de solución que involucren principios de matemáticas e ingeniería, buscando siempre minimizar la posibilidad de que intrusos accedan a la información de forma fraudulenta.

Palabras clave: criptografía, esteganografía, seguridad informática, atractores caóticos, cuaterniones, autómatas celulares.

¿Cómo citar este libro?

Alvarado Nieto, D., Amaya Barrera, I. y Suárez Parra, C. A. (2024). *Una propuesta criptográfica basada en atractores caóticos*. Universidad Distrital Francisco José de Caldas-Editorial UD. <https://doi.org/10.69740/UD.9789587877090.9789587877113>

Abstract

This book is the result of a collaborative academic exercise involving students and professors from the Systems Engineering program at the Universidad Distrital Francisco José de Caldas. Together, we aimed to apply the principles of chaotic dynamic systems in the context of computer security. To achieve this, we addressed topics such as chaotic attractors, quaternion theory, and cellular automata, which facilitate handling large amounts of information and generating cryptographic and steganographic proposals from this perspective. The models presented here were applied to various large files such as images, audio, texts, videos, and multimedia files. They underwent security and performance evaluations and achieved indicators consistent with those reported in current scientific literature, demonstrating that such proposals could eventually be implemented in real environments. Given the rapid advancement of technologies, information security has become a significant social problem with a global impact. This fact motivates the academic community to work interdisciplinarily to lead robust and effective solution proposals involving mathematics and engineering principles. The primary aim is always to minimize the possibility of intruders accessing information fraudulently.

Keywords: cryptography, steganography, computer security, chaotic attractors, quaternions, cellular automata.

How to cite this book?

Alvarado Nieto, D., Amaya Barrera, I., & Suárez Parra, C. A. (2024). *Una propuesta criptográfica basada en atractores caóticos*. Universidad Distrital Francisco José de Caldas-Editorial UD. <https://doi.org/10.69740/UD.9789587877090.9789587877113>

Los autores agradecen el apoyo recibido por parte de la Oficina de Investigaciones de la Universidad Distrital Francisco José de Caldas para la ejecución del proyecto de investigación que dio origen a este libro.

Asimismo, resaltan la vinculación de estudiantes de Ingeniería de Sistemas de la Universidad Distrital Francisco José de Caldas al grupo de investigación ComplexUD, quienes contribuyeron significativamente a la implementación de los algoritmos desarrollados para llevar a cabo las pruebas de cada uno de los modelos propuestos.



© Universidad Distrital Francisco José de Caldas
© Oficina de Investigaciones
© Deicy Alvarado Nieto, Isabel Amaya Barrera,
César Suárez Parra (autores)

ISBN: 978-958-787-709-0
ISBN digital: 978-958-787-710-6
ISBN ePub: 978-958-787-711-3

DOI: 10.69740/
UD.9789587877090.9789587877113

Primera edición, septiembre de 2024

Líder Unidad de Publicaciones
Rubén Eliécer Carvajalino C.

Gestión editorial
Rosa Isabel González Moreno

Corrección de estilo
HIPERTEXTO SAS

Diagramación
HIPERTEXTO SAS

Diseño de cubierta
Astrid Prieto Castillo

Editorial UD

Universidad Distrital Francisco José de Caldas
Carrera 24 n.º 34-37 Bogotá, D. C., Colombia
Teléfono: 6013239300 ext. 6202
Correo electrónico: publicaciones@udistrital.edu.co

Cláusula de responsabilidad:

Editorial UD no es responsable de las opiniones y de la información recogidas en el presente texto. Los autores asumen de manera exclusiva y plena toda responsabilidad sobre su contenido, ya sea este propio o de terceros, declarando en este último supuesto que cuentan con la debida autorización de terceros para su publicación. Igualmente, expresan que no existe conflicto de interés alguno en relación con los resultados de la investigación propiedad de tales terceros. En consecuencia, los autores serán responsables civil, administrativa o penalmente, frente a cualquier reclamo o demanda por parte de terceros, relativa a los derechos de autor u otros derechos que se vulneren como resultado de su contribución.

Todos los derechos reservados.

Esta obra no puede ser reproducida sin el permiso previo escrito de la Unidad de Publicaciones de la Universidad Distrital Francisco José de Caldas.
Hecho en Colombia.

Sistema de Bibliotecas de la Universidad Distrital Francisco José de Caldas Catalogación en la publicación (CEP)

Alvarado Nieto, Deicy

Una propuesta criptográfica basada en atractores caóticos / Deicy Alvarado Nieto, Isabel Amaya Barrera, César Suárez Parra. -- Primera edición. -- Bogotá : Universidad Distrital Francisco José de Caldas, 2024.

149 páginas : ilustraciones, gráficas ; 24 cm. (Espacios)

ISBN: 978-958-787-709-0 ISBN digital: 978-958-787-710-6
ISBN ePub: 978-958-787-711-3

1. Análisis matemático 2. Criptografía 3. Cifrado de datos -- Matemáticas 4. Cifrado de datos -- Informática 5. Sistemas dinámicos diferenciales 6. Comportamiento caótico en sistemas I. Amaya Barrera, Isabel, autora II. Suárez Parra, César, autor III. Serie.

515.39: CDD 21 edición.

Contenido

Introducción	13
Capítulo 1. Caos y atractores extraños	15
Reseña histórica	15
Fundamentación teórica	16
Características	25
Referencias	29
Capítulo 2. Principios de la teoría de cuaterniones	31
Reseña histórica	31
Fundamentación teórica	32
Exponencial de números cuaterniones	40
Características	43
Transformada discreta de Fourier de cuaterniones	57
Referencias	62
Capítulo 3. Autómatas celulares	65
Reseña histórica	65
Fundamentación teórica	66
Características de los autómatas celulares	76
Tipos de autómatas	78
Referencias	84
Capítulo 4. Principios de esteganografía	87
Reseña histórica	87
Fundamentación teórica	89
Estegoanálisis	101
Referencias	105

Capítulo 5. Criptografía	109
Reseña histórica	109
Características	110
Tipos de criptografía	111
Referencias	117
Capítulo 6. Nuevas propuestas criptográficas	121
Reseña histórica	121
Motivación	122
Aplicación de atractores caóticos en imágenes a color	123
Aplicación de la teoría de cuaterniones	129
Aplicación de autómatas celulares	133
Aplicación de técnicas de esteganografía	139
Aplicación de algoritmos genéticos	142
Referencias	146
Sobre los autores	149

Introducción

El Grupo de Complejidad de la Universidad Distrital ComplexUD es reconocido por su trayectoria en el campo de las ciencias de la complejidad. Desde el año 2006 viene organizando periódicamente el Encuentro Interuniversitario de Complejidad, lo que ha permitido consolidar una comunidad académica interesada en esta área a nivel nacional e internacional. Dentro de sus líneas de investigación se encuentran: Comportamientos Colectivos, Informática Educativa, Logística Hospitalaria, Problemas de Transporte y Sistemas Complejos. En estos últimos se enmarcan aquellos sistemas que presentan comportamientos caóticos y una de sus aplicaciones se enfoca en la seguridad de la información. Con base en esto, se propuso ante la Oficina de Investigaciones de la Universidad Distrital Francisco José de Caldas el desarrollo del proyecto de investigación *Modelos de encriptación basados en atractores caóticos*, el cual vinculó a estudiantes de pregrado de Ingeniería de Sistemas de la Facultad de Ingeniería de la misma Universidad y permitió obtener resultados académicos significativos.

Al interior del proyecto, motivados por la necesidad de mejorar eficientemente la seguridad de las comunicaciones confidenciales y aprovechando las características inherentes a los sistemas caóticos, se plantearon nuevos esquemas de encriptación con indicadores de velocidad y seguridad acordes con los publicados en la literatura reciente. Estos se complementaron recurriendo a teoría de autómatas celulares, algoritmos genéticos y técnicas esteganográficas. Cabe destacar que con el desarrollo del proyecto se lograron publicar tres artículos en revistas científicas y se

llevaron a cabo tres proyectos de grado en modalidad de investigación, lo cual permitió la graduación de seis de los estudiantes vinculados al proyecto.

El desarrollo de proyectos, como el que dio origen a este libro, es de vital importancia dado que el avance de la tecnología ha incrementado de forma exponencial el intercambio de información a través de canales digitales y, a la vez, ha debilitado la seguridad de dicha información. Coyunturalmente, la pandemia provocada por la COVID-19 ha disparado aún más la utilización de medios digitales para la comunicación y el intercambio de información sensible, lo que evidencia que este tipo de ejercicios académicos cobran mayor preponderancia en una situación de confinamiento, como la que vivió el mundo durante el 2020 y el 2021.

Este libro se presenta con el fin de divulgar parte del trabajo llevado a cabo por el Grupo de Complejidad ComplexUD en relación con la seguridad de la información usando sistemas complejos. El libro está estructurado en seis capítulos: en los primeros cinco capítulos se describe la fundamentación teórica sobre el caos, los atractores extraños, los cuaterniones, los autómatas celulares, la criptografía y la esteganografía, y en el capítulo seis se esquematizan nuevos modelos de seguridad aplicados a imágenes, videos, texto y sonido, utilizando las temáticas abordadas en los capítulos iniciales.

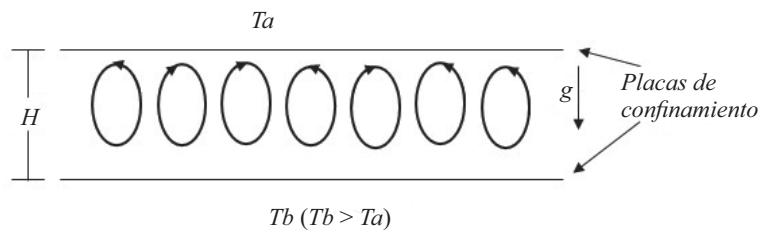
Capítulo 1. Caos y atractores extraños

Reseña histórica

La formulación matemática para modelar el proceso de convección de calor en la atmósfera, llevada a cabo por el meteorólogo Edward Norton Lorenz en 1963, fue pionera en el estudio de la teoría del caos. En dicho estudio, Lorenz evidenció un fenómeno bastante extraño e inexplicable: cuando alimentaba su modelo redondeando cifras decimales, conseguía resultados totalmente diferentes a cuando utilizaba las cifras originales, lo que conllevaba a generar predicciones climáticas completamente diferentes. Si bien los datos eran muy cercanos inicialmente y los resultados de las iteraciones del modelo matemático parecían semejantes, rápidamente se obtenían soluciones totalmente diferentes. En la actualidad y en el contexto de la teoría del caos, este fenómeno se conoce como sensibilidad a condiciones iniciales (Suárez Parra, 2020).

Dado que las capas en un fluido caliente suben debido a su menor densidad, se genera un proceso de transferencia de calor cuando hay convección natural, situación que permite tener una mayor comprensión del modelo de Lorenz (Suárez Parra, 2020). En la figura 1 se ilustra el proceso de convección y se muestran los vórtices o remolinos del fluido, donde H es la altura de la capa de fluido considerada, T_a y T_b son las temperaturas de las placas de confinamiento que indican los límites de la capa de fluido entre los cuales se analiza la transferencia de calor y g es la aceleración de la gravedad (Campos Romero e Isaza Delgado, 2002).

Figura 1. Convección natural para generar transferencia de calor



Fuente: Suárez Parra (2020).

La formulación del modelo de Lorenz se sintetiza por medio del sistema de ecuaciones (1).

$$\begin{aligned}\frac{dx}{dt} &= \sigma(y - x) \\ \frac{dy}{dt} &= rx - y - xz \\ \frac{dz}{dt} &= xy - bz\end{aligned}\tag{1}$$

La variable x indica la velocidad y el sentido de circulación del fluido, si $x > 0$, el fluido circula en sentido horario y, si $x < 0$, el fluido circula en sentido antihorario. Las variables y y z hacen referencia a las variaciones horizontales y verticales de la temperatura, respectivamente. El parámetro σ , conocido como número de Prandtl, describe la viscosidad cinemática (ν) respecto a la conductividad térmica $R(\alpha) = \sigma = \nu/\alpha$ y b define la relación existente entre el tamaño horizontal de los vórtices de convección y la altura de la capa de fluido. R es el número de Rayleigh y está dado por la expresión (2), R_c es el valor de R para el cual comienza el proceso de convección también llamado número crítico de Rayleigh y r corresponde al cociente entre estos dos valores: $r = \frac{R}{R_c}$. Cabe resaltar que los parámetros σ , b y r son adimensionales (Suárez Parra, 2020).

$$R = \frac{g\alpha H^3 (T_b - T_a)}{\nu k} \geq 0 \quad T_b > T_a\tag{2}$$

En la expresión (2), g es la aceleración de la gravedad, α es el coeficiente de expansión térmica, H es la altura de la capa de fluido considerada, T_b y T_a son las temperaturas de las placas de confinamiento de dicha capa, ν es la viscosidad cinemática y k es la conductividad térmica (Campos Romero e Isaza Delgado, 2002; Lorenz, 1963).

Fundamentación teórica

El concepto matemático de caos que se analiza a continuación fue propuesto por Robert Luke Devaney en 1986 (Avellán Zapata, 2015; Balibrea Gallego, 1999; Devaney, 1989).

Dado el conjunto X , $f: X \rightarrow X$ o $\dot{x} = f(x)$, $x \in X$, dependiendo de si se analiza en forma discreta o continua, respectivamente, el sistema es caótico sobre X si cumple las siguientes condiciones:

1. f tiene dependencia sensible a las condiciones iniciales.
2. f es topológicamente transitiva.
3. El conjunto de puntos periódicos es denso en X .

Un sistema dinámico $f: X \rightarrow X$ o $\dot{x} = f(x)$ tiene dependencia sensible a las condiciones iniciales si existe un $\delta > 0$, tal que, para todo $x \in X$ y para todo $\varepsilon > 0$, existe un $y \in X$ con distancia $d(x, y) < \varepsilon$ y, para todo $n_0 \geq 0$ o $t \geq 0$, se cumple $d(f^{n_0}x, f^{n_0}y) > \delta$ o $d(f(t, x), f(t, y)) > \delta$.

El sistema $f: X \rightarrow X$ o $\dot{x} = f(x)$ es topológicamente transitivo si para cualquier par de conjuntos abiertos y no vacíos $U, V \subset X$ existe un $n_0 \geq 0$ o $t_0 \geq 0$ tal que $f^{n_0}(U) \cap V \neq \emptyset$ o $f(t_0, U) \cap V \neq \emptyset$. Lo anterior significa que el sistema dinámico no se puede separar en dos subconjuntos abiertos disyuntos.

Un subconjunto U de X es denso si su clausura $\bar{U} = X$. La clausura de U consiste en todos los puntos de U junto con todos los puntos límites de U si U es abierto (Devaney, 1989).

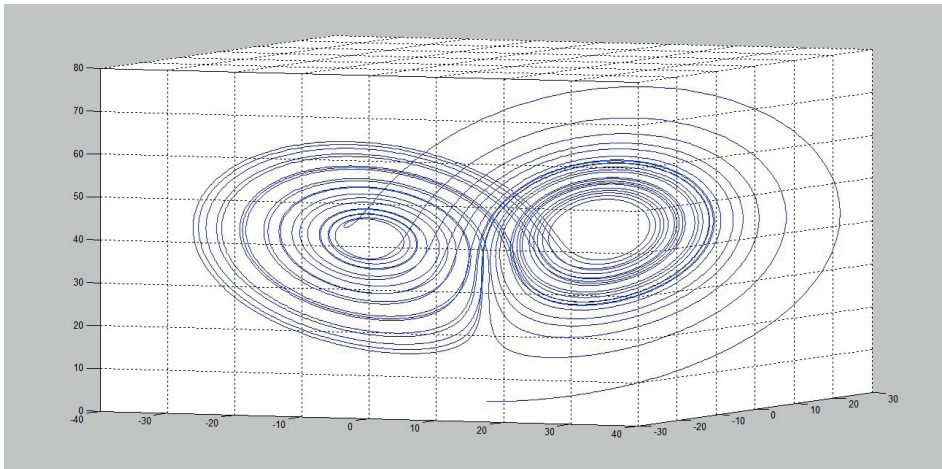
El conjunto cerrado A se denomina atractor si cumple las siguientes condiciones (Alligood *et al.*, 1997; Bahi y Guyeux, 2013; Kanti Bala *et al.*, 2017; Strogatz, 1994):

1. Es invariante, lo cual significa que, para cada $x \in A$ y $n \geq 0$, $f^n(x) \in A$ o para todo $t \geq 0$, $f(t, x) \in A$, es decir, cualquier trayectoria inicia en A y permanece en A todo el tiempo.
2. Atrae un conjunto abierto de condiciones iniciales que están suficientemente cercanas a él. Si hay un conjunto abierto U que contiene a A , tal que $x(0) \in U$, entonces, la distancia $d(x(t), A)$ tiende a cero cuando $t \rightarrow \infty$ o la distancia $d(x(n), A)$ tiende a cero cuando $n \rightarrow \infty$ si el sistema se analiza en forma continua o discreta, respectivamente. El conjunto A más grande de condiciones iniciales que pueda ser atraído por A se denomina la cuenca de atracción o base de atracción del conjunto A .

3. Es minimal, es decir, no hay un subconjunto propio de A que satisfaga las condiciones 1 y 2.

Un atractor extraño es aquel que presenta dependencia sensitiva a las condiciones iniciales (Strogatz, 1994). En la figura 2 se muestra el atractor extraño que se genera con el sistema de ecuaciones de Lorenz para los parámetros $\sigma = 10$, $r = 40$ y $b = 3$, el cual se obtuvo mediante el método de integración numérica Runge-Kutta de cuarto orden (Blanes Zamora *et al.*, 2020; Ricardo, 2008).

Figura 2. Atractor de Lorenz



Fuente: Suárez Parra (2020).

Un indicador que permite medir la sensibilidad de los atractores a las condiciones iniciales tiene que ver con los exponentes de Lyapunov, los cuales se describen a continuación.

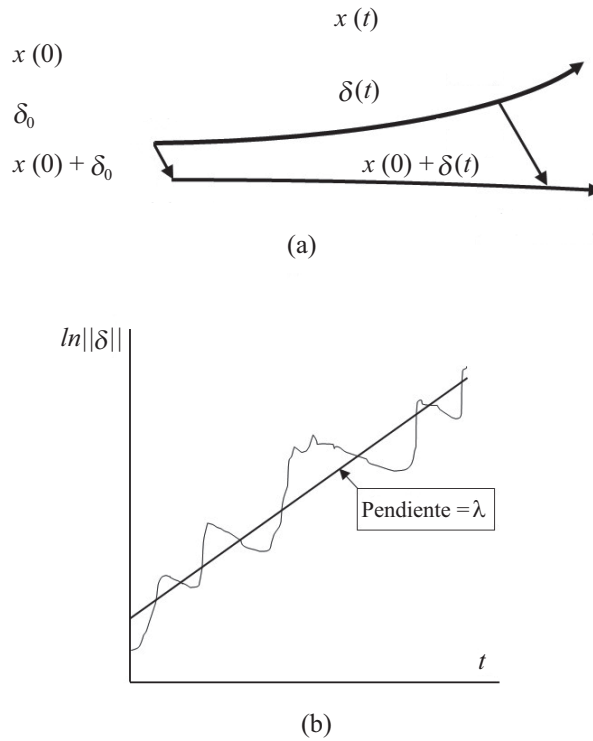
Dado el sistema $\dot{x} = f(x)$, al analizar sus trayectorias a partir de dos condiciones iniciales muy cercanas separadas vectorialmente una distancia $\delta_{(0)}$, como se muestra en la figura 3(a), se ha encontrado que, para un

tiempo t , la aproximación de la separación vectorial entre las dos trayectorias $\delta_{(t)}$ está dada por la expresión (3).

$$\|\delta_{(t)}\| \sim \|\delta_{(0)}\| e^{\lambda t} \quad (3)$$

$\|\cdot\|$ denota la norma vectorial, λ se denomina el exponente de Lyapunov y corresponde a la pendiente de la gráfica del $\ln \|\delta_{(t)}\|$ contra t , como se ilustra en la figura 3(b) (Strogatz, 1994).

Figura 3. Análisis del exponente de Lyapunov (tiempo continuo)



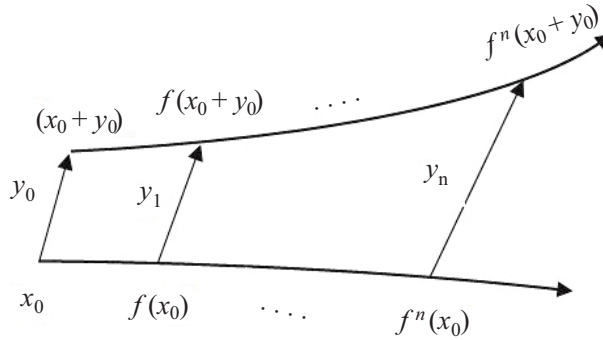
Fuente: Suárez Parra (2020).

Formalmente, en tiempo discreto, el exponente de Lyapunov se define a partir del número de Lyapunov (Olmedo *et al.*, 2007) de la siguiente forma:

Si se considera el sistema $x_{k+1} = f(x_k)$ y se hace evolucionar a partir de dos condiciones iniciales muy cercanas: x_0 , $x_0 + y_0$, tal como se evidencia en la figura 4, vectorialmente se tiene la expresión (4).

$$f(x_0 + y_0) = f(x_0) + y_1 \quad (4)$$

Figura 4. Análisis del exponente de Lyapunov (tiempo discreto)



Fuente: Suárez Parra (2020).

Dividiendo la expresión (4) entre y_0 y aplicando el concepto de derivada, se obtiene la expresión (5).

$$\frac{y_1}{y_0} = \lim_{y_0 \rightarrow 0} \frac{f(x_0 + y_0) - f(x_0)}{y_0} = f'(x_0) \quad (5)$$

De lo cual se deduce la expresión (6).

$$|y_1| = |f'(x_0)| |y_0| \quad (6)$$

Aplicando reiteradamente este proceso, se llega a la expresión (7).

$$\begin{aligned}
|y_2| &= |f'(x_1)| |y_1|, |y_2| = |f'(x_1)| |f'(x_0)| |y_0| \\
|y_3| &= |f'(x_2)| |y_2|, |y_3| = |f'(x_2)| |f'(x_1)| |f'(x_0)| |y_0| \\
&\vdots \\
|y_n| &= |f'(x_{n-1})| |y_{n-1}|, |y_n| = |f'(x_{n-1})| |f'(x_{n-2})| \cdots |f'(x_0)| |y_0| \\
\frac{|y_n|}{|y_0|} &= |f'(x_{n-1})| |y_{n-1}|, |y_n| = |f'(x_{n-1})| |f'(x_{n-2})| \cdots |f'(x_0)|
\end{aligned} \tag{7}$$

Siendo f una función derivable con derivada continua sobre la recta real $\mathbb{R}$, el número de Lyapunov $L(x_0)$ de la órbita $\{x_0, x_1, x_2, x_3, \dots\}$ es definido por medio de la expresión (8) si el límite existe.

$$L(x_0) = \lim_{n \rightarrow \infty} \left(|f'(x_0)| \cdots |f'(x_{n-1})| \right)^{\frac{1}{n}} \tag{8}$$

El exponente de Lyapunov está dado por la expresión (9) si el límite existe.

$$\lambda(x_0) = \lim_{n \rightarrow \infty} \frac{1}{n} \left[\ln |f'(x_0)| + \dots + \ln |f'(x_{n-1})| \right] \tag{9}$$

Otra forma de calcular los exponentes de Lyapunov es mediante los autovalores en la expresión (10), lo cual es útil cuando se analizan sistemas multidimensionales.

$$\lambda_j(x_0) = \lim_{n \rightarrow \infty} \frac{1}{2n} \ln(H_{nj}); j = 1, 2, \dots, k \tag{10}$$

H_{jn} son los autovalores de la matriz $H_n(x_0)$ dada por las expresiones (11) y (12).

$$H_n(x_0) = \left(D_f^n(x_{(0)}) \right)^T \cdot D_f^n(x_{(0)}) \tag{11}$$

$$D_f^n(x_0) = D_f(x_{n-1}) \cdot D_f(x_{n-2}), \dots, D_f(x_0) \tag{12}$$

Cada uno de los $D_f(x_i)$ corresponde a la matriz jacobiana del sistema, evaluada en los puntos solución de las iteradas y en las condiciones iniciales.

Ejemplo 1

Al calcular el exponente de Lyapunov para el atractor de Lorenz con $\sigma = 10$, $r = 40$, $b = 3$ y condiciones iniciales $(0.1, 0.1, 0.1)$ y $(0.2, 0.2, 0.2)$ mediante el método de integración numérica Runge-Kutta de cuarto orden, se implementó en Matlab el programa identificado como Código 1, el cual inicia indicando el incremento de tiempo que se maneja en cada iteración, los valores de los parámetros y las condiciones iniciales de los dos atractores de Lorenz utilizados, y calculando la distancia inicial entre ellos. Luego, se implementa el método de integración numérica Runge Kutta de cuarto orden para diez mil iteraciones, calculando en cada iteración los valores de las variables de los dos atractores y su distancia. Después del proceso iterativo, se implementa el método de mínimos cuadrados para hallar la recta de regresión del valor absoluto del logaritmo natural de las distancias entre los atractores y la pendiente de esta recta, que es el exponente de Lyapunov. Finalmente, se codifican las instrucciones de ploteo, donde se puede elegir entre graficar las trayectorias de los atractores o graficar el valor absoluto del logaritmo natural de las distancias y su recta de regresión.

En resumen, el Código 1 permite: encontrar la trayectoria seguida por el atractor para cada condición inicial, determinar la distancia entre las dos trayectorias en cada iteración, hallar la recta de regresión por mínimos cuadrados del valor absoluto del logaritmo natural de las diferencias, calcular la pendiente de la recta de regresión y, con ello, calcular el exponente de Lyapunov, $\lambda = 0.1922$ con diez mil iteraciones, tal como se evidencia en las figuras 5 y 6.

Código 1. Cálculo de los exponentes de Lyapunov

```
clear all;
close all;
h=0.001;x1=0.1;x2=0.1;x3=0.1;xi1=0.1;xi2=0.1;xi3=0.1;t(1)=0;n=1;
a=10;b=3;r=40;
u1=0.2;u2=0.2;u3=0.2;
```

```

ui1=0.2;ui2=0.2;ui3=0.2;
st(n)=t(1);
d(1)=abs(log(sqrt((xi1-ui1)^2+(xi2-ui2)^2+(xi3-
ui3)^2)));sd(n)=d(1);
ptd(1)=t(1)*d(1);sptd(n)=ptd(1);tq(1)=(t(1))^2;stq(n)=tq(1);
y(:,1)=[0.1;0.1;0.1];
y1(:,1)=[0.2;0.2;0.2];

for k=1:10000
    k1=h*a*(x2-x1);
    p1=h*a*(u2-u1);
    m1=h*(r*x1-x2-x1*x3);
    q1=h*(r*u1-u2-u1*u3);
    r1=h*(x1*x2-b*x3);
    s1=h*(u1*u2-b*u3);
    k2=h*a*((x2+(m1/2))-(x1+(k1/2)));
    p2=h*a*((u2+(q1/2))-(u1+(p1/2)));
    m2=h*(r*(x1+(k1/2))-(x2+(m1/2))-((x1+(k1/2))*(x3+(r1/2))));
    q2=h*(r*(u1+(p1/2))-(u2+(q1/2))-((u1+(p1/2))*(u3+(s1/2))));
    r2=h*((x1+(k1/2))*(x2+(m1/2))-b*(x3+(r1/2)));
    s2=h*((u1+(p1/2))*(u2+(q1/2))-b*(u3+(s1/2)));
    k3=h*a*((x2+(m2/2))-(x1+(k2/2)));
    p3=h*a*((u2+(q2/2))-(u1+(p2/2)));
    m3=h*(r*(x1+(k2/2))-(x2+(m2/2))-((x1+(k2/2))*(x3+(r2/2))));
    q3=h*(r*(u1+(p2/2))-(u2+(q2/2))-((u1+(p2/2))*(u3+(s2/2))));
    r3=h*((x1+(k2/2))*(x2+(m2/2))-b*(x3+(r2/2)));
    s3=h*((u1+(p2/2))*(u2+(q2/2))-b*(u3+(s2/2)));
    k4=h*a*((x2+m3)-(x1+k3));
    p4=h*a*((u2+q3)-(u1+p3));
    m4=h*(r*(x1+k3)-(x2+m3)-((x1+k3)*(x3+r3)));
    q4=h*(r*(u1+p3)-(u2+q3)-((u1+p3)*(u3+s3)));
    r4=h*((x1+k3)*(x2+m3)-b*(x3+r3));
    s4=h*((u1+p3)*(u2+q3)-b*(u3+s3));
    x1=x1+(1/6)*(k1+2*k2+2*k3+k4);
    u1=u1+(1/6)*(p1+2*p2+2*p3+p4);
    x2=x2+(1/6)*(m1+2*m2+2*m3+m4);
    u2=u2+(1/6)*(q1+2*q2+2*q3+q4);
    x3=x3+(1/6)*(r1+2*r2+2*r3+r4);
    u3=u3+(1/6)*(s1+2*s2+2*s3+s4);
    n=n+1;
    t(n)=(n-1)*h;
    st(n)=st(n-1)+t(n);

```

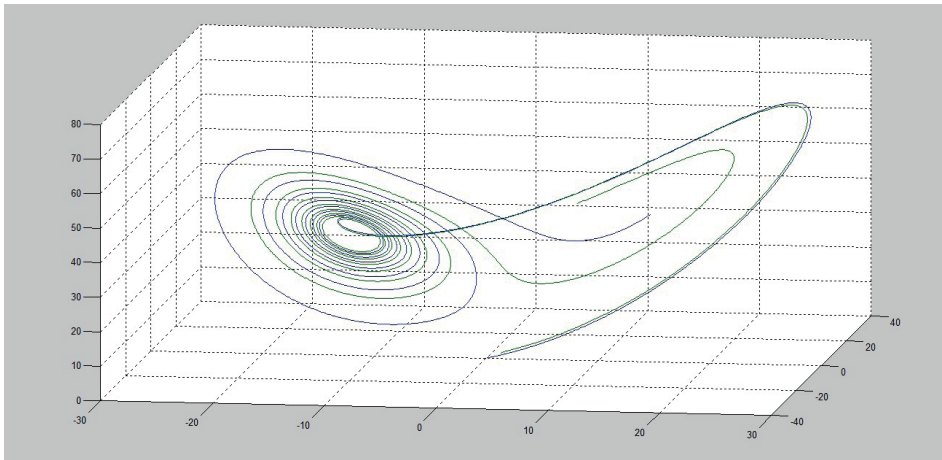
```

y(:,n)=[x1,x2,x3];
y1(:,n)=[u1,u2,u3];
d(n)=abs(log(sqrt((x1-u1)^2+(x2-u2)^2+(x3-u3)^2)));
ptd(n)=t(n)*d(n);tq(n)=(t(n))^2;stq(n)=stq(n-1)+tq(n);
sd(n)=sd(n-1)+d(n);sptd(n)=sptd(n-1)+ptd(n);
end

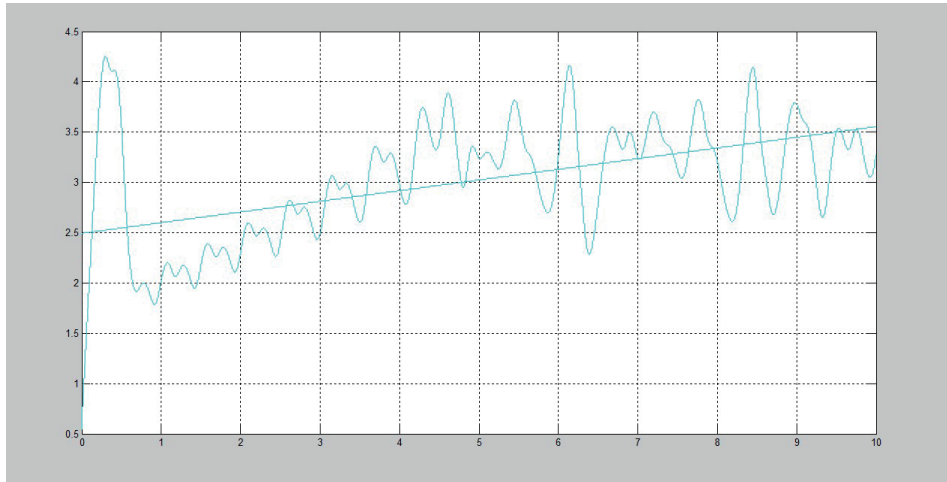
m=((sptd(n)-(st(n)*sd(n)/n))/(stq(n)-(st(n))^2/n));
c=(sd(n)/n)-(m*st(n)/n);
t(1)=0;
k=1;
Y(1)=m*t(1)+c;
for i=2:(n)
    k=k+1;
    t(k)=(k-1)*h;
    Y(i)=m*t(k)+c;
end
%plot(t,d,'c',t,Y,'c')
plot3(y(1,:),y(2,:),y(3,:),y1(1,:),y1(2,:),y1(3,:))
%plot3(y1(1,:),y1(2,:),y1(3,:))
grid

```

Figura 5. Trayectorias para el atractor de Lorenz con $\sigma = 10$, $r = 40$, $b = 3$ y condiciones iniciales $(0.1, 0.1, 0.1)$ y $(0.2, 0.2, 0.2)$



Fuente: Suárez Parra (2020).

Figura 6. Exponente de Lyapunov, $\lambda = 0.1922$ con 10000 iteraciones

Fuente: Suárez Parra (2020).

Características

Una de las ventajas que tienen los sistemas dinámicos caóticos hace referencia a la teoría de sincronización. La sincronización de sistemas caóticos es el proceso mediante el cual se puede lograr que dos o más de estos sistemas logren un comportamiento dinámico común después de un estado transitorio. Este principio de sincronización fue descubierto por Louis M. Pecora y Thomas L. Carroll, quienes demostraron que, al descomponer los sistemas caóticos en subsistemas que presenten todos sus exponentes de Lyapunov negativos y encadenarlos con una o más señales comunes, los subsistemas tomarán el mismo comportamiento que el original. Cuando se desean sincronizar dos sistemas caóticos, se toma uno como maestro o conductor y el otro como esclavo o seguidor; además, se debe elegir una variable de enlace entre los dos sistemas, la cual se denomina variable sincronizante o variable conductora (Cuomo *et al.*, 1993; García Montes, 2018; Lugo Peñaloza, 2016; Pecora y Carroll, 1990).

Ejemplo 2

Se ilustra la sincronización del atractor de Lorenz con dos subsistemas de este. Considerando el sistema original de Lorenz como conductor, la variable

sincronizante con el primer subsistema será x y la variable sincronizante entre el primer y el segundo subsistema será y_1 . Los subsistemas 1 y 2 se definen por medio de las expresiones (13) y (14), respectivamente.

$$\begin{aligned}\dot{x}_1 &= \sigma(y_1 - x) \\ \dot{z}_1 &= xy_1 - bz_1\end{aligned}\tag{13}$$

$$\begin{aligned}\dot{y}_2 &= rx_2 - y_1 - x_2z_2 \\ \dot{z}_2 &= x_2y_1 - bz_2\end{aligned}\tag{14}$$

Como se indicó anteriormente, para que los subsistemas se sincronicen con el original, los exponentes de Lyapunov de los subsistemas deben ser negativos. Esto significa que, para el subsistema 1, $|y_1 - y|$ y $|z_1 - z| \rightarrow 0$ cuando $t \rightarrow \infty$. De forma semejante, para el subsistema 2, $|x_2 - x|$ y $|z_2 - z| \rightarrow 0$ cuando $t \rightarrow \infty$.

Con el fin de apreciar la sincronización, se implementó en Matlab un programa identificado como Código 2, donde inicialmente se definen el incremento de tiempo en cada iteración, el valor de los parámetros y las condiciones iniciales del sistema conductor de Lorenz y de los dos subsistemas considerados. Luego, se implementa el método de integración numérica Runge Kutta de cuarto orden con tres mil iteraciones; en cada iteración, se halla el valor de las variables del sistema conductor y de los dos subsistemas, y estos resultados se guardan en forma matricial. Posteriormente, una vez terminado el proceso iterativo, se codifican las instrucciones para graficar las trayectorias de los tres atractores, el conductor y los dos subsistemas, como se muestra en las figuras 7, 8 y 9.

Código 2. Sincronización de atractores caóticos

```
clear all;
close all;
h=0.001;x1=0.1;x2=0.1;x3=0.1;xi1=30.1;u2=20.2;xi3=30.1;t(1)=0;n=1;
a=10;b=3;r=40;
x1=0.1;u2=20.2;u3=20.2;
st(n)=t(1);
```

```

y(:,1)=[0.1;0.1;0.1];
Y1(:,1)=[0.1;20.2;20.2];
Y2(:,1)=[30.1;20.2;30.1];
for k=1:3000
    k1=h*a*(x2-x1);
    p1=h*a*(u2-x1);
    i1=h*a*(u2-xi1);
    m1=h*(r*x1-x2-x1*x3);
    q1=h*(r*x1-u2-x1*u3);
    j1=h*(r*xi1-u2-xi1*xi3);
    r1=h*(x1*x2-b*x3);
    s1=h*(x1*u2-b*u3);
    v1=h*(xi1*u2-b*xi3);
    k2=h*a*((x2+(m1/2))-(x1+(k1/2)));
    p2=h*a*((u2+(q1/2))-(x1+(p1/2)));
    i2=h*a*((u2+(j1/2))-(xi1+(i1/2)));
    m2=h*(r*(x1+(k1/2))-(x2+(m1/2))-((x1+(k1/2))*(x3+(r1/2))));
    q2=h*(r*(x1+(p1/2))-(u2+(q1/2))-((x1+(p1/2))*(u3+(s1/2))));
    j2=h*(r*(xi1+(i1/2))-(u2+(j1/2))-((xi1+(i1/2))*(xi3+(v1/2))));
    r2=h*((x1+(k1/2))*(x2+(m1/2))-b*(x3+(r1/2)));
    s2=h*((x1+(p1/2))*(u2+(q1/2))-b*(u3+(s1/2)));
    v2=h*((xi1+(i1/2))*(u2+(j1/2))-b*(xi3+(v1/2)));
    k3=h*a*((x2+(m2/2))-(x1+(k2/2)));
    p3=h*a*((u2+(q2/2))-(x1+(p2/2)));
    i3=h*a*((u2+(j2/2))-(xi1+(i2/2)));
    m3=h*(r*(x1+(k2/2))-(x2+(m2/2))-((x1+(k2/2))*(x3+(r2/2))));
    q3=h*(r*(x1+(p2/2))-(u2+(q2/2))-((x1+(p2/2))*(u3+(s2/2))));
    j3=h*(r*(xi1+(i2/2))-(u2+(j2/2))-((xi1+(i2/2))*(xi3+(v2/2))));
    r3=h*((x1+(k2/2))*(x2+(m2/2))-b*(x3+(r2/2)));
    s3=h*((x1+(p2/2))*(u2+(q2/2))-b*(u3+(s2/2)));
    v3=h*((xi1+(i2/2))*(u2+(j2/2))-b*(xi3+(v2/2)));
    k4=h*a*((x2+m3)-(x1+k3));
    p4=h*a*((u2+q3)-(x1+p3));
    i4=h*a*((u2+j3)-(xi1+i3));
    m4=h*(r*(x1+k3)-(x2+m3)-((x1+k3)*(x3+r3)));
    q4=h*(r*(x1+p3)-(u2+q3)-((x1+p3)*(u3+s3)));
    j4=h*(r*(xi1+i3)-(u2+j3)-((xi1+i3)*(xi3+v3)));
    r4=h*((x1+k3)*(x2+m3)-b*(x3+r3));
    s4=h*((x1+p3)*(u2+q3)-b*(u3+s3));
    v4=h*((xi1+i3)*(u2+j3)-b*(xi3+v3));
    x1=x1+(1/6)*(k1+2*k2+2*k3+k4);
    %x1=x1+(1/6)*(p1+2*p2+2*p3+p4);
    xi1=xi1+(1/6)*(i1+2*i2+2*i3+i4);

```

```

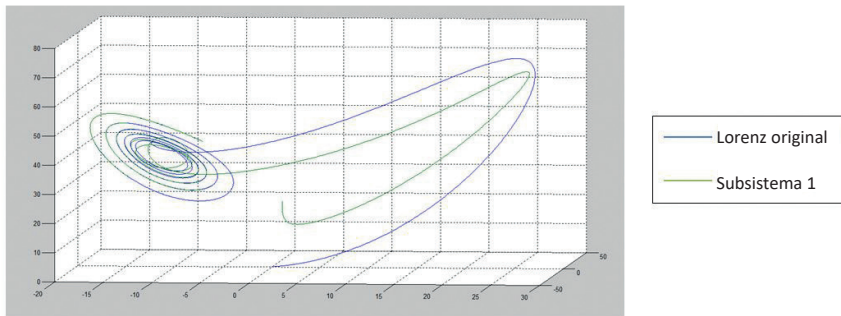
x2=x2+(1/6)*(m1+2*m2+2*m3+m4);
u2=u2+(1/6)*(q1+2*q2+2*q3+q4);
%u2=u2+(1/6)*(j1+2*j2+2*j3+j4);
x3=x3+(1/6)*(r1+2*r2+2*r3+r4);
u3=u3+(1/6)*(s1+2*s2+2*s3+s4);
xi3=xi3+(1/6)*(v1+2*v2+2*v3+v4);
n=n+1;
t(n)=(n-1)*h;
Y(:,n)=[x1,x2,x3];
Y1(:,n)=[x1,u2,u3];
Y2(:,n)=[xi1,u2,xi3];
end

plot3(Y(1,:),Y(2,:),Y(3,:),Y(1,:),Y1(2,:),Y1(3,:),Y2(1,:),
Y2(2,:),Y2(3,:))

grid

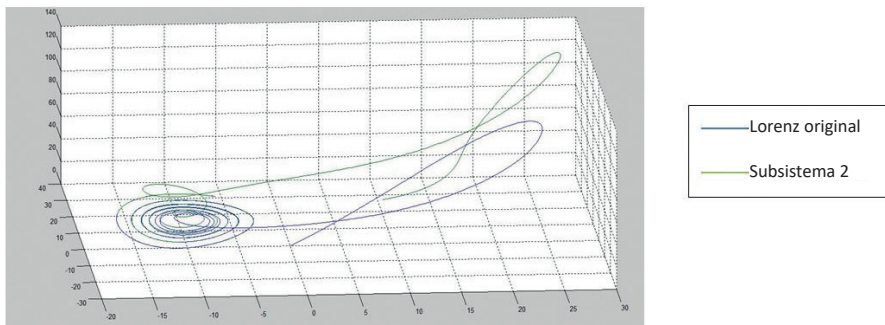
```

Figura 7. Sincronización entre el sistema original de Lorenz y el subsistema 1



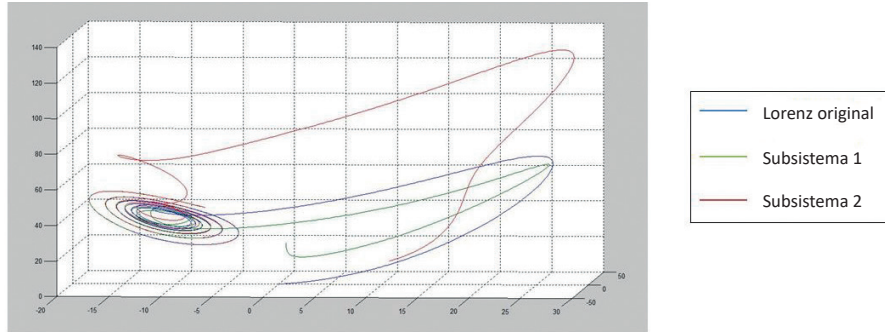
Fuente: Suárez Parra (2020).

Figura 8. Sincronización entre el sistema original de Lorenz y el subsistema 2



Fuente: Suárez Parra (2020).

Figura 9. Sincronización entre el sistema original de Lorenz y los subsistemas 1 y 2



Fuente: Suárez Parra (2020).

Varios de los trabajos que se describen en el capítulo 6 de este libro se basan en la teoría de sincronismo y atractores caóticos.

Referencias

- Alligood, K. T., Sauer, T. D. y Yorke, J. A. (1997). *Chaos: An Introduction to Dynamical Systems*. Springer-Verlag.
- Avellán Zapata, V. (2015). *Caos, linealidad y dimensión* [Trabajo de grado, Universidad de Murcia]. Archivo digital. https://www.um.es/documents/118351/1884002/TFG_ABELLAN+ZAPATA.pdf/5209b4f3-63ab-4274-9160-996c735ff197
- Bahi, J. y Guyeux, C. (2013). *Discrete Dynamical Systems and Chaotic Machines: Theory and Applications* (1.^a ed.). CRC Press. <https://doi.org/10.1201/b14979>
- Balibrea Gallego, F. (1999). Caos y atractores extraños. Dos problemas no lineales en matemáticas. *La Gaceta*, 2(1), 99-116. <https://gaceta.rsme.es/abrir.php?id=368>
- Blanes Zamora, S., Ginestar Peiró, D. y Roselló Ferragud, M. D. (2020). *Introducción a los métodos numéricos para ecuaciones diferenciales* (3.^a ed.). Universitat Politècnica de València. https://gdocu.upv.es/alfresco/service/api/node/content/workspace/SpacesStore/9267df62-d8ab-46e3-849f-e2a9bafa0be9/TOC_6100_01_03.pdf?guest=true
- Campos Romero, D. e Isaza Delgado, J. F. (2002). *Prolegómenos a los sistemas dinámicos*. Universidad Nacional de Colombia.

- Cuomo, K. M., Oppenheim, A. V. y Strogatz, S. H. (1993). Synchronization of Lorenz-based chaotic circuits with applications to communications. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 40(10), 626-633. <https://doi.org/10.1109/82.246163>
- Devaney, R. L. (1989). *An Introduction to Chaotic Dynamical Systems* (2.^a ed.). Addison-Wesley Publishing Company. https://zangeneh.iut.ac.ir/sites/zangeneh.iut.ac.ir/files/files_course/robert_1.devaneyan_introduciponto_chaotic_dynamical_system.pdf
- García Montes, E. (2018). *Sincronización de sistemas vía acoplamientos dinámicos* [Tesis de maestría, Centro de Investigación Científica y de Educación Superior de Ensenada, Baja California]. CICESE. <http://cicese.repositorioinstitucional.mx/jspui/handle/1007/2524>
- Kanti Bala, B., Mohamed Arshad, F. y Mohd Noh, K. (2017). *System Dynamics: Modelling and Simulation* (1.^a ed.). Springer.
- Lorenz, E. N. (1963). Deterministic Nonperiodic Flow. *Journal of the Atmospheric Sciences*, 20(2), 130-141. [https://doi.org/10.1175/1520-0469\(1963\)020<0130:DNF>2.0.CO;2](https://doi.org/10.1175/1520-0469(1963)020<0130:DNF>2.0.CO;2)
- Lugo Peñaloza, A. F. (2016). *Preservación de sincronización en sistemas dinámicos fraccionarios* [Tesis doctoral, Universidad Iberoamericana]. IBERO. <https://ri.iberomx/bitstream/handle/iberomx/467/016166s.pdf?sequence=1&isAllowed=y>
- Olmedo, E., Gimeno, R., Escot, L. y Mateos, R. (2007). Convergencia y estabilidad de los tipos de cambio europeos: una aplicación de exponentes de Lyapunov. *Cuadernos de Economía*, 44(129), 91-108. <http://doi.org/10.4067/S0717-68212007000100004>
- Pecora, L. M. y Carroll, T. L. (1990). Synchronization in Chaotic Systems. *Physical Review Letters*, 64(8), 821-824. <https://doi.org/10.1103/PhysRevLett.64.821>
- Ricardo, H. (2008). *Ecuaciones diferenciales: una introducción moderna* (1.^a ed.). Editorial Reverté.
- Strogatz, S. H. (1994). *Nonlinear Dynamics and Chaos: with Applications to Physics, Biology, Chemistry, and Engineering*. Perseus Books Publishing. <http://users.uoa.gr/~pjioannou/nonlin/Strogatz,%20S.%20H.%20-%20Nonlinear%20Dynamics%20And%20Chaos.pdf>
- Suárez Parra, C. A. (2020, 16 de abril). *Cibernética cualitativa II - Notas de clase*. Course Hero. <https://www.coursehero.com/file/59860059/Caos-y-Atractores-ca%C3%B3icos-2-Notas-de-clasepdf/>

Capítulo 2. Principios de la teoría de cuaterniones

Reseña histórica

El físico matemático irlandés William Hamilton es recordado en la ciencia por sus aportes en los campos de la mecánica, la óptica y la astronomía. Alrededor de 1840 se encontraba interesado en hallar una extensión del conjunto de números complejos a más de dos dimensiones. Como consecuencia de este esfuerzo, formuló la teoría de números cuaterniones, que resultó de gran interés para la comunidad científica de la época, ya que serían el lenguaje adecuado para representar el espacio tiempo y permitirían el desarrollo de nuevas teorías sobre dicho conjunto (Sánchez Muñoz, 2011).

Aunque inicialmente los cuaterniones estuvieron un poco rezagados debido al desarrollo del cálculo, se logró definir la estructura algebraica de manera formal. Esto corroboró propiedades que se han utilizado en aplicaciones de otras áreas de la ciencia y de la ingeniería. Algunas de estas aplicaciones se han desarrollado en electromagnetismo, mecánica cuántica, robótica, visión por ordenador, tratamiento de imágenes a color y modelos de encriptación (ElSharkawy, 2017; Kamlofsky y Bergamini, 2015; Mahecha, 2008).

Una de las primeras ventajas que se detectó en este conjunto tiene que ver con la aplicación en el entorno de las rotaciones, por lo cual resulta útil para ubicación espacial, aplicaciones virtuales y animación en tres dimensiones. Esto es posible gracias a que, al realizar una implementación computacional, esta es más eficiente porque se reduce tanto el espacio de almacenamiento requerido como la cantidad de operaciones a realizar, en comparación con el procedimiento habitual en términos de matrices; aunque se presentan

ambigüedades al hacer referencia a rotaciones sucesivas, esta situación se elimina con el uso de los cuaterniones. Estas ventajas han sido aprovechadas en aplicaciones sobre el control de la orientación de aeronaves, submarinos y naves espaciales. Algunas referencias que pueden resultar útiles para obtener información más detallada sobre la teoría de cuaterniones son Altmann (1989), Bannon (2015) y Kapoor *et al.* (2021).

Fundamentación teórica

Dados cuatro números reales q_1, q_2, q_3, q_4 , el cuaternión Q se define por medio de la expresión (15).

$$Q = q_1 + q_2i + q_3j + q_4k \quad (15)$$

Siendo i, j, k los vectores de la base canónica de $\mathbb{R}^3$, el término q_1 se denomina parte escalar del cuaternión Q y la expresión $q_2i + q_3j + q_4k$ es la parte vectorial del cuaternión Q . De esta forma, los números reales se conciben como cuaterniones con parte vectorial cero y los vectores de $\mathbb{R}^3$ como cuaterniones con parte real cero, también llamados cuaterniones puros.

En analogía con el conjunto de los números complejos, se tiene que, si los vectores i, j, k se asumen como si fuesen unidades imaginarias, el conjunto de números cuaterniones es también conocido como conjunto de números hipercomplejos de rango cuatro. Si se establecen las igualdades dadas en (16), se deduce que el conjunto $\{1, i, j, k\}$ es una base para $\mathbb{R}^4$.

$$\begin{aligned} 1 &= (1, 0, 0, 0) \\ i &= (0, 1, 0, 0) \\ j &= (0, 0, 1, 0) \\ k &= (0, 0, 0, 1) \end{aligned} \quad (16)$$

El conjunto de cuaterniones suele notarse con la letra H . Se dice que dos cuaterniones P y Q , donde $P = p_1 + p_2i + p_3j + p_4k$ y $Q = q_1 + q_2i + q_3j + q_4k$, son iguales si y solo si: $p_1 = q_1, p_2 = q_2, p_3 = q_3$, y $p_4 = q_4$.

A continuación, se presentan algunos conceptos similares a los definidos en el conjunto de números complejos.

Dado el cuaternión de la expresión (17):

$$Q = q_1 + q_2i + q_3j + q_4k \quad (17)$$

Se definen el cuaternión conjugado $\bar{Q}$ y el módulo de Q por medio de las expresiones (18) y (19), respectivamente.

$$\bar{Q} = q_1 - q_2i - q_3j - q_4k \quad (18)$$

$$\|Q\| = \sqrt[2]{q_1^2 + q_2^2 + q_3^2 + q_4^2} \quad (19)$$

Un cuaternión con módulo unidad se denomina cuaternión unitario. Este es de interés cuando se hace referencia a la representación de rotaciones.

Un cuaternión unitario $Q = q_1 + q_2i + q_3j + q_4k$ se representa a través de la expresión (20).

$$Q = \cos \phi + v \sin \phi = (\cos \phi, \sin \phi \cdot q_2, \sin \phi \cdot q_3, \sin \phi \cdot q_4) \quad (20)$$

$v = (q_2, q_3, q_4)$ es un vector unitario de $\mathbb{R}^3$ y ϕ es un ángulo en el intervalo $[0, 2\pi)$. En estas condiciones, el cuaternión Q es unitario efectivamente, como se muestra en la expresión (21).

$$Q = \sqrt{\cos^2(\phi) + \sin^2(\phi)q_2^2 + \sin^2(\phi)q_3^2 + \sin^2(\phi)q_4^2} = 1 \quad (21)$$

El conjunto de cuaterniones con módulo 1 forma la esfera tridimensional denotada como S^3 en el espacio cuatridimensional.

Representación vectorial y matricial de los cuaterniones

En notación matemática, el conjunto de números cuaterniones H se enuncia a través de (22):

$$H = \{q_1 + q_2i + q_3j + q_4k; q_1, q_2, q_3, q_4 \in \mathbb{R}\} \subseteq \mathbb{R}^4 \quad (22)$$

O de forma equivalente, por medio de (23):

$$H = \{(q_1 + q_2i) + (q_3 + q_4i)j : q_1 + q_2i, q_3 + q_4i \in \mathbb{C}\} \subseteq \mathbb{C}^2 \quad (23)$$

Este tipo de notación se conoce como representación vectorial de los cuaterniones, aunque también se pueden describir como se muestra en la expresión (24).

$$Q = s(Q) + v(Q) \quad (24)$$

$s(Q)$ y $v(Q)$ denotan la parte escalar y vectorial de Q . De forma alternativa, el cuaternión $Q = q_1 + q_2i + q_3j + q_4k$ se puede expresar como el producto interno de dos vectores, de los cuales uno es q_1, q_2, q_3, q_4 y el otro $(1, i, j, k)$.

Otra manera válida de representar cuaterniones es utilizando el conjunto de matrices complejas 2x2, denotado como $M_2(\mathbb{C})$. Sea A el conjunto dado en (25):

$$A = \left\{ \begin{pmatrix} a & b \\ -\bar{b} & \bar{a} \end{pmatrix} : a, b \in \mathbb{C} \right\} \subset M_2(\mathbb{C}) \quad (25)$$

A cada cuaternión $P = p_1 + p_2i + p_3j + p_4k$ se le hace corresponder una matriz del conjunto A , como se especifica en (26):

$$\begin{pmatrix} p_1 + p_2i & p_3 + p_4i \\ -(p_3 - p_4i) & p_1 - p_2i \end{pmatrix} \quad (26)$$

La función que define esta correspondencia es biyectiva; los conjuntos A y H resultan ser anillos isomorfos.

En esta biyección, a los cuaterniones básicos $1, i, j$ y k se les asignan las correspondencias dadas en (27):

$$1 \rightarrow \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad i \rightarrow \begin{pmatrix} i & 0 \\ 0 & -i \end{pmatrix} \quad j \rightarrow \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \quad k \rightarrow \begin{pmatrix} 0 & i \\ i & 0 \end{pmatrix} \quad (27)$$

De esta forma, se muestra que los cuaterniones pueden ser representados como matrices 2×2 con entradas complejas. Nótese que el determinante de la matriz que representa el cuaternión P coincide con el módulo al cuadrado de P . De manera similar, usando matrices reales 4×4 , el cuaternión Q se puede representar por medio de (28):

$$A = \begin{pmatrix} q_1 & q_2 & q_3 & q_4 \\ -q_2 & q_1 & -q_4 & q_3 \\ -q_3 & q_4 & q_1 & -q_2 \\ -q_4 & -q_3 & q_2 & q_1 \end{pmatrix} \quad (28)$$

Lo anterior se puede reescribir como se muestra en (29):

$$A = q_1 \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} + q_2 \begin{pmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 \end{pmatrix} + q_3 \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix} + q_4 \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{pmatrix} \quad (29)$$

Estas últimas cuatro matrices representan en su orden la correspondencia entre los cuaterniones básicos $1, i, j, k$.

Además, el determinante de la matriz A asociada al cuaternión Q está dado por la expresión (30).

$$\det A = (q_1^2 + q_2^2 + q_3^2 + q_4^2)^2 = ||Q||^4 \quad (30)$$

Álgebra de cuaterniones

Se definen las operaciones de suma, resta y multiplicación por un escalar a través de las expresiones (31) y (32).

La suma de cuaterniones cumple las propiedades clausurativa, conmutativa, asociativa, modulativa y existencia de elemento inverso aditivo; todas estas propiedades son heredadas del espacio vectorial $\mathbb{R}^n$.

$$P \pm Q = (p_1 \pm q_1) + i(p_2 \pm q_2) + j(p_3 \pm q_3) + k(p_4 \pm q_4) \quad (31)$$

La multiplicación de un escalar por un cuaternión se define como en el caso de $\mathbb{R}^n$, es decir, si $\alpha \in \mathbb{R}$, el cuaternión $\alpha \cdot P$ corresponde a (32):

$$\alpha \cdot P = \alpha p_1 + \alpha p_2 i + \alpha p_3 j + \alpha p_4 k \quad (32)$$

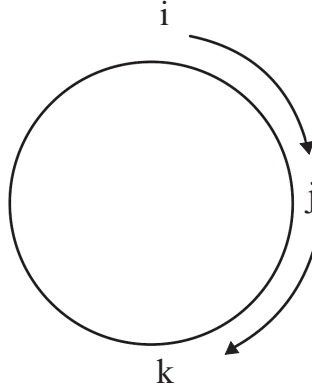
Para definir la operación multiplicación, Hamilton introdujo los productos especiales entre las unidades i , j y k , los cuales se describen en la tabla 1.

Tabla 1. Productos especiales

	i	j	k
i	-1	k	$-j$
j	$-k$	-1	i
k	j	$-i$	-1

Fuente: elaboración propia.

Los productos especiales se obtienen disponiendo i , j y k en el sentido de las manecillas del reloj y los signos negativos aparecen cuando se multiplica en el sentido contrario a la dirección de las flechas de la figura 10.

Figura 10. Regla para la multiplicación de cuaterniones

Fuente: elaboración propia.

De esta forma, se define la operación producto cuaternión como se especifica en la expresión (33).

$$PQ = (p_1 + p_2i + p_3j + p_4k) \cdot (q_1 + q_2i + q_3j + q_4k) \quad (33)$$

Esta ecuación se puede reescribir por medio de la expresión (34).

$$PQ = p_1q_1 + ip_1q_2 + jp_1q_3 + kp_1q_4 + ip_2q_1 - p_2q_2 + kp_2q_3 - jp_2q_4 + jp_3q_1 - kp_3q_2 + kp_4q_1 - jp_4q_2 - ip_4q_3 - p_4q_4 - p_3q_3 + ip_3q_4 \quad (34)$$

A partir de la expresión (34), se deduce la igualdad (35).

$$||Q|| = (Q \cdot \bar{Q})^{\frac{1}{2}} \quad (35)$$

En términos de las partes real y vectorial de P y Q , el cuaternión PQ se puede expresar así por medio de la expresión (36) (Somma, 2018).

$$PQ = s(P)s(Q) - v(P) \cdot v(Q) + s(P)v(Q) + s(Q)v(P) + v(P) \times v(Q) \quad (36)$$

En el caso en que P y Q sean cuaterniones puros, se deducen las expresiones (37) y (38).

$$PQ = -v(P) \cdot v(Q) + v(P) \times v(Q) = -P \cdot Q + P \times Q \quad (37)$$

$$QP = -v(Q) \cdot v(P) + v(Q) \times v(P) = -Q \cdot P + Q \times P = -P \cdot Q - P \times Q \quad (38)$$

Los símbolos $\cdot$ y $\times$ denotan el producto punto y vectorial definidos en $\mathbb{R}^3$, respectivamente.

Con esto, se obtienen las expresiones (39) y (40).

$$P \times Q = \frac{1}{2}(PQ - QP) \quad (39)$$

$$PQ + P \cdot Q = \frac{1}{2}(PQ - QP) \quad (40)$$

De donde se deduce la igualdad (41).

$$P \cdot Q = \frac{-1}{2}(PQ + QP) \quad (41)$$

La multiplicación de cuaterniones es clausurativa, asociativa y distributiva con respecto a la operación suma, pero no es conmutativa. H junto con las operaciones suma y producto satisface las propiedades de campo, excepto que el producto no es conmutativo. Además, de manera análoga al campo de los números complejos, se demuestra que la norma del cuaternión PQ es el producto de las normas de los cuaterniones P y Q , como se indica en la expresión (42).

$$\|PQ\| = \|P\| \|Q\| \quad (42)$$

En particular, se deduce que el producto de números cuaterniones unitarios es de nuevo un cuaternión unitario. Utilizando como norma el módulo

definido para cuaterniones, se observa que H forma un álgebra de Banach real (Rubio, 2019).

De forma análoga al caso de los números complejos. se hace alusión al inverso multiplicativo de un cuaternión. Para todo cuaternión P no nulo, se define el inverso multiplicativo de P a través de la expresión (43).

$$P^{-1} = \frac{\bar{P}}{\|P\|^2} \quad (43)$$

La multiplicación entre un cuaternión y su inverso satisface la propiedad de conmutatividad, como se indica en la expresión (44).

$$PP^{-1} = P^{-1}P = 1 \quad (44)$$

Como caso particular se tiene que, si el cuaternión es unitario, su inverso es justamente el conjugado, tal como se presenta en la expresión (45).

$$P^{-1} = \bar{P} \quad (45)$$

A partir de lo anterior, se define el cociente entre los cuaterniones P y Q , con $Q \neq 0$, por medio de las expresiones (46) y (47).

$$\frac{P}{Q} = PQ^{-1} = \frac{P\bar{Q}}{\|Q\|^2} \quad (46)$$

$$Q^{-1}P = \frac{\bar{Q}P}{\|Q\|^2} \quad (47)$$

Otro concepto que se define en el contexto de cuaterniones es la traza. Dado el cuaternión $P = p_1 + p_2i + p_3j + p_4k$, la traza de P denotada por $tr(P)$ es dada por la expresión (48).

$$tr(P) = P + \bar{P} = 2p_1 \quad (48)$$

A partir de la traza, se define una función bilineal simétrica que cumple las propiedades de espacio euclidiano bajo el producto interior definido por f , la cual está determinada por las expresiones (49) y (50).

$$f: H \times H \rightarrow H \quad (49)$$

$$f(P, Q) = \frac{1}{2} \text{tr}[P\bar{Q}] = \frac{1}{2}(P\bar{Q} + \overline{PQ}) = \frac{1}{2}(P\bar{Q} + Q\bar{P}) = p_1q_1 + p_2q_2 + p_3q_3 + p_4q_4 \quad (50)$$

Además, H resulta isomorfo a $\mathbb{R}^4$. A partir del producto interno, se puede hacer alusión al concepto de ortogonalidad y ortonormalidad de cuaterniones.

La función distancia definida en (51) es una métrica para los cuaterniones. Las operaciones de suma y multiplicación definidas sobre estos presentan una estructura de anillo de división no conmutativo.

$$d(P, Q) = \|Q\| - P \quad \forall P, Q \in H \quad (51)$$

Exponencial de números cuaterniones

Como se mencionó previamente, un cuaternión unitario $Q = q_1 + q_2i + q_3j + q_4k$ se puede representar como se indica en la expresión (52).

$$Q = \cos \phi + v \sin \phi = (\cos(\phi), \sin(\phi) \cdot q_2, \sin(\phi) \cdot q_3, \sin(\phi) \cdot q_4) \quad (52)$$

$v = (q_2, q_3, q_4)$ es un vector unitario de $\mathbb{R}^3$ y ϕ es un ángulo en el intervalo $[0, 2\pi)$. Dicha expresión se obtiene haciendo una analogía con el caso de los números complejos definido en la expresión (52). Con el ángulo $\phi \in [0, \pi]$, se verifican las expresiones (53) y (54).

$$\cos \phi = \frac{q_1}{\sqrt{q_1^2 + q_2^2 + q_3^2 + q_4^2}} \quad (53)$$

$$\text{sen}\phi = \frac{\sqrt{q_2^2 + q_3^2 + q_4^2}}{\sqrt{q_1^2 + q_2^2 + q_3^2 + q_4^2}} \quad (54)$$

La identidad de Euler se puede generalizar para números cuaterniones puros unitarios. Si $Q = q_2i + q_3j + q_4k$ es unitario, para el vector $v = (q_2, q_3, q_4)$ en $\mathbb{R}^3$ de norma 1 y $\phi \in [0, \pi]$, se define $e^{v\phi}$ por medio de la expresión (55).

$$e^{v\phi} = \cos\phi + v \cdot \text{sen}\phi = Q \quad (55)$$

De forma similar al caso complejo, se define la potencia de un cuaternión unitario. Sea $Q = \cos\phi + v \cdot \text{sen}\phi$ un cuaternión unitario, siendo $v = (q_2, q_3, q_4)$ en $\mathbb{R}^3$ de norma 1 y $\phi \in [0, \pi]$, para $t \in \mathbb{R}$, se define el cuaternión Q^t por medio de la expresión (56).

$$Q^t = (\cos\phi + v \cdot \text{sen}\phi)^t = \cos(t\phi) + v \text{sen}(t\phi) \quad (56)$$

Para el cuaternión $Q = q_1 + q_2i + q_3j + q_4k$, se define la exponencial cuaternión en la expresión (57).

$$e^Q = e^{q_1 + q_2i + q_3j + q_4k} = e^{q_1 + v} = e^{q_1} e^v, \quad v = (q_2, q_3, q_4) \quad (57)$$

Si $\check{v}$ denota el cuaternión unitario obtenido a partir de v , entonces, $v = \check{v} \|v\|$. De esta forma, se obtiene la expresión (58).

$$e^Q = e^{q_1} e^{\check{v} \|v\|} = e^{q_1} (\cos\|v\| + \check{v} \cdot \text{sen}\|v\|) \quad (58)$$

Esta ecuación equivale a la expresión (59).

$$e^Q = e^{q_1} e^{\check{v}\|\check{v}\|} = e^{q_1} \left(\cos\|\check{v}\| + \check{v} \cdot \text{sen}\|\check{v}\| \right) \quad (59)$$

De lo anterior se deduce la expresión (60).

$$e^Q = e^{q_1} \left(\cos\sqrt{q_2^2 + q_3^2 + q_4^2} + \frac{v}{\sqrt{q_2^2 + q_3^2 + q_4^2}} \cdot \text{sen}\sqrt{q_2^2 + q_3^2 + q_4^2} \right) \quad (60)$$

Esta ecuación se puede reescribir como la expresión (61).

$$e^Q = e^{q_1} \left(\cos\sqrt{q_2^2 + q_3^2 + q_4^2} + v \cdot \frac{\text{sen}\sqrt{q_2^2 + q_3^2 + q_4^2}}{\sqrt{q_2^2 + q_3^2 + q_4^2}} \right) \quad (61)$$

Por consiguiente, se deduce la expresión (62).

$$e^Q = e^{q_1} \left(\cos\sqrt{q_2^2 + q_3^2 + q_4^2} + \frac{\text{sen}\sqrt{q_2^2 + q_3^2 + q_4^2}}{\sqrt{q_2^2 + q_3^2 + q_4^2}} (q_2 i + q_3 j + q_4 k) \right) \quad (62)$$

Como caso particular, si el cuaternión Q es unitario con parte imaginaria no nula, utilizando la expresión (63):

$$q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1 \quad (63)$$

Se concluye la expresión (64).

$$\sqrt{q_2^2 + q_3^2 + q_4^2} = \sqrt{1 - q_1^2} \quad (64)$$

Si se satisface la condición $1 - q_1^2 \geq 0$, se tiene que $q_1 \in (-1, 1)$.

En tal caso, se tiene la expresión (65).

$$e^Q = e^{q_1} \left(\cos \sqrt{1 - q_1^2} + \frac{\sin \sqrt{1 - q_1^2}}{\sqrt{1 - q_1^2}} (q_2 i + q_3 j + q_4 k) \right) \quad (65)$$

Características

La importancia de los cuaterniones unitarios se centra en que a través de estos se pueden representar rotaciones en tres dimensiones de manera más simple que cuando se hacen en términos de vectores en $\mathbb{R}^3$. Algunas referencias que se pueden consultar para ampliar este apartado son Bergamini y Kamlofsky (2015) y Somma (2018).

Rotaciones con números cuaterniones

Los cuaterniones se pueden utilizar para representar rotaciones en el espacio tridimensional (Cova, 2019). Las matrices ortogonales de tamaño 3×3 son las que sirven para tal estudio. Una matriz cuadrada A de tamaño n se considera ortogonal si $A^{-1} = A^T$. Este tipo de matrices satisfacen que su determinante sea ± 1 y tienen una estructura de grupo con la multiplicación de matrices.

En el caso particular en que A sea ortogonal de tamaño 2 y determinante 1, se deben satisfacer las ecuaciones (66) y (67).

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad (66)$$

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad (67)$$

A partir de estas, se obtienen las igualdades dadas en (68).

$$a_{11}^2 + a_{12}^2 = 1 \quad a_{21}a_{11} + a_{22}a_{12} = 0 \quad a_{21}^2 + a_{22}^2 = 1 \quad (68)$$

El hecho de que $a_{11}^2 + a_{12}^2 = 1$ garantiza la existencia de un ángulo $\theta \in [0, 2\pi)$ tal que se cumplen las igualdades dadas en (69).

$$a_{11} = \cos\theta; a_{12} = \sen\theta \quad (69)$$

Además, como $a_{11}a_{21} + a_{12}a_{22} = 0$, entonces, $a_{11}a_{21} = -a_{12}a_{22}$. Si $a_{11} \neq 0$, $a_{21} = \frac{-a_{12}a_{22}}{a_{11}}$ y con $\rho = \frac{a_{22}}{a_{11}}$ se obtiene la expresión (70).

$$a_{21} = -\rho a_{12} = -\rho \sen\theta \quad (70)$$

A partir de esta, se obtienen las expresiones (71) y (72).

$$a_{21}^2 = \rho^2 \sen^2\theta \quad (71)$$

$$a_{22}^2 = \rho^2 a_{11}^2 = \rho^2 \cos^2\theta \quad (72)$$

Al sumar las ecuaciones (71) y (72), se obtiene la expresión (73).

$$a_{21}^2 + a_{22}^2 = \rho^2 = 1 \quad (73)$$

Esto implica que $\rho = \pm 1$. Si $\rho = 1$, entonces, se cumple (74):

$$a_{21} = -\sen\theta \text{ y } a_{22} = a_{11} \quad (74)$$

De esta manera, la matriz A toma la forma (75):

$$A = \begin{pmatrix} \cos\theta & \sen\theta \\ -\sen\theta & \cos\theta \end{pmatrix} \quad (75)$$

Si A se interpreta como la matriz de una transformación lineal en el plano, se deduce que esta representa una rotación de un ángulo θ en sentido contrario a las manecillas del reloj.

Ahora, suponiendo que $\rho = -1$, se obtiene (76):

$$a_{21} = a_{12} \quad a_{11} = -a_{22} \quad (76)$$

Con esto, la matriz A resulta de la forma (77):

$$A = \begin{pmatrix} \cos\theta & \sin\theta \\ \sin\theta & -\cos\theta \end{pmatrix} \quad (77)$$

En este caso, A representa una rotación de un ángulo θ seguida de una simetría con relación al eje x .

Para el caso de una rotación en $\mathbb{R}^3$, el propósito es determinar la forma de la matriz que represente una rotación de un ángulo de θ radianes.

El propósito es girar un vector $\tilde{u} \neq 0$ en $\mathbb{R}^3$ alrededor de un eje en la dirección de un vector unitario v un ángulo de θ radianes en sentido positivo. Con este fin, se considera $\tilde{u} = (u_1, u_2, u_3) \in \mathbb{R}^3$, el cual se puede representar como el cuaternión $u = (0, u_1, u_2, u_3)$. De otra parte, se considera el cuaternión $Q = \cos\left(\frac{\theta}{2}\right) + \sin\left(\frac{\theta}{2}\right)v$ al calcular el producto cuaternión $u\bar{Q}$. Al utilizar la expresión (36), se obtiene la expresión (78).

$$\begin{aligned} u\bar{Q} &= u \left[\cos\left(\frac{\theta}{2}\right) - \sin\left(\frac{\theta}{2}\right)v \right] = (0 + \tilde{u}) \cos\left(\frac{\theta}{2}\right) - \sin\left(\frac{\theta}{2}\right)v \\ &= -(u_1, u_2, u_3) \cdot \left(-\sin\left(\frac{\theta}{2}\right)v + \cos\left(\frac{\theta}{2}\right)(u_1, u_2, u_3) + (u_1, u_2, u_3)X \left(-\sin\left(\frac{\theta}{2}\right)v \right) \right) \\ &= -\tilde{u} \left(-\sin\left(\frac{\theta}{2}\right)v \right) + \cos\left(\frac{\theta}{2}\right)\tilde{u} - \tilde{u}X \sin\left(\frac{\theta}{2}\right)v \\ &= \sin\left(\frac{\theta}{2}\right)\tilde{u} \cdot v + \cos\left(\frac{\theta}{2}\right)\tilde{u} - \sin\left(\frac{\theta}{2}\right)\tilde{u} \times v \end{aligned} \quad (78)$$

Al emplear esta última expresión y calcular $Qu\bar{Q}$, se obtiene la expresión (79).

$$Qu\bar{Q} = \left(\cos\left(\frac{\theta}{2}\right) + \operatorname{sen}\left(\frac{\theta}{2}\right)v \right) \left(\operatorname{sen}\left(\frac{\theta}{2}\right)\tilde{u} \cdot v + \cos\left(\frac{\theta}{2}\right)\tilde{u} - \operatorname{sen}\left(\frac{\theta}{2}\right)\tilde{u} \times v \right) \quad (79)$$

A partir de esta, se llega a la igualdad (80).

$$\begin{aligned} Qu\bar{Q} = & \cos\left(\frac{\theta}{2}\right)\operatorname{sen}\left(\frac{\theta}{2}\right)\tilde{u} \cdot v - \operatorname{sen}\left(\frac{\theta}{2}\right)v \cdot \left(\cos\left(\frac{\theta}{2}\right)\tilde{u} - \operatorname{sen}\left(\frac{\theta}{2}\right)\tilde{u} \times v \right) + \\ & \cos\left(\frac{\theta}{2}\right) \left(\cos\left(\frac{\theta}{2}\right)\tilde{u} - \operatorname{sen}\left(\frac{\theta}{2}\right)\tilde{u} \times v \right) + \left(\operatorname{sen}\left(\frac{\theta}{2}\right)\tilde{u} \cdot v \right) \operatorname{sen}\left(\frac{\theta}{2}\right)v + \\ & \operatorname{sen}\left(\frac{\theta}{2}\right)v \times \left(\cos\left(\frac{\theta}{2}\right)\tilde{u} - \operatorname{sen}\left(\frac{\theta}{2}\right)\tilde{u} \times v \right) \end{aligned} \quad (80)$$

Esta se puede reescribir por medio de la expresión (81).

$$\begin{aligned} Qu\bar{Q} = & \cos\left(\frac{\theta}{2}\right)\operatorname{sen}\left(\frac{\theta}{2}\right)\tilde{u} \cdot v - \operatorname{sen}\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right)v \cdot \tilde{u} + \operatorname{sen}^2\left(\frac{\theta}{2}\right)v \cdot (\tilde{u} \times v) + \\ & \cos^2\left(\frac{\theta}{2}\right)\tilde{u} - \cos\left(\frac{\theta}{2}\right)\operatorname{sen}\left(\frac{\theta}{2}\right)\tilde{u} \times v + \operatorname{sen}^2\left(\frac{\theta}{2}\right)(\tilde{u} \cdot v)v + \\ & \operatorname{sen}\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right)v \times \tilde{u} - \operatorname{sen}^2\left(\frac{\theta}{2}\right)v \times (\tilde{u} \times v) \end{aligned} \quad (81)$$

Por tanto, se obtiene la expresión (82).

$$\begin{aligned} Qu\bar{Q} = & \operatorname{sen}^2\left(\frac{\theta}{2}\right)v \cdot (\tilde{u} \times v) + \cos^2\left(\frac{\theta}{2}\right)\tilde{u} - \cos\left(\frac{\theta}{2}\right)\operatorname{sen}\left(\frac{\theta}{2}\right)\tilde{u} \times v + \\ & \operatorname{sen}^2\left(\frac{\theta}{2}\right)(\tilde{u} \cdot v)v + \operatorname{sen}\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right)v \times \tilde{u} - \operatorname{sen}^2\left(\frac{\theta}{2}\right)v \times (\tilde{u} \times v) \end{aligned} \quad (82)$$

Esta es equivalente a la expresión (83).

$$\begin{aligned}
Qu\bar{Q} = \cos^2\left(\frac{\theta}{2}\right)u - \cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\theta}{2}\right)u \times v + \sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right)v \times u + \\
\sin^2\left(\frac{\theta}{2}\right)(u \cdot v)v - \sin^2\left(\frac{\theta}{2}\right)v \times (u \times v)
\end{aligned} \quad (83)$$

A partir de esta, se obtiene la expresión (84).

$$Qu\bar{Q} = \sin^2\left(\frac{\theta}{2}\right)(\tilde{u} \cdot v)v + \cos^2\left(\frac{\theta}{2}\right)\tilde{u} + 2\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right)v \times \tilde{u} - \sin^2\left(\frac{\theta}{2}\right)v \times (\tilde{u} \times v) \quad (84)$$

Esta puede sintetizarse por medio de las expresiones (85) y (86).

$$Qu\bar{Q} = \sin^2\left(\frac{\theta}{2}\right)(\tilde{u} \cdot v)v + \cos^2\left(\frac{\theta}{2}\right)\tilde{u} - \sin^2\left(\frac{\theta}{2}\right)v \times (\tilde{u} \times v) + \sin(\theta)v \times \tilde{u} \quad (85)$$

$$Qu\bar{Q} = \sin^2\left(\frac{\theta}{2}\right)(\tilde{u} \cdot v)v + \cos^2\left(\frac{\theta}{2}\right)\tilde{u} - \sin^2\left(\frac{\theta}{2}\right)[v \cdot v\tilde{u} - \tilde{u} \cdot vv] + \sin(\theta)v \times \tilde{u} \quad (86)$$

Finalmente, se llega a la expresión (87).

$$\begin{aligned}
Qu\bar{Q} &= \sin^2\left(\frac{\theta}{2}\right)(\tilde{u} \cdot v)v + \cos^2\left(\frac{\theta}{2}\right)\tilde{u} - \sin^2\left(\frac{\theta}{2}\right)[\tilde{u} - \tilde{u} \cdot vv] + \sin(\theta)v \times \tilde{u} \\
&= \sin^2\left(\frac{\theta}{2}\right)(\tilde{u} \cdot v)v + \cos^2\left(\frac{\theta}{2}\right)\tilde{u} - \sin^2\left(\frac{\theta}{2}\right)\tilde{u} + \sin^2\left(\frac{\theta}{2}\right)\tilde{u} \cdot vv + \sin(\theta)v \times \tilde{u} \\
&= 2\sin^2\left(\frac{\theta}{2}\right)(\tilde{u} \cdot v)v + \left(\cos^2\left(\frac{\theta}{2}\right) - \sin^2\left(\frac{\theta}{2}\right)\right)\tilde{u} + \sin(\theta)v \times \tilde{u} \\
&= (1 - \cos\theta)(\tilde{u} \cdot v)v + \cos\theta\tilde{u} + \sin(\theta)v \times \tilde{u}
\end{aligned} \quad (87)$$

La expresión del lado derecho muestra el resultado de rotar el vector $\tilde{u}$ un ángulo de θ radianes en sentido positivo alrededor de un eje en la dirección de un vector unitario v . Lo anterior se sustenta mediante la fórmula de rotación de Rodrigues (Somma, 2018).

Del desarrollo anterior se concluye que, dado un vector $\tilde{u} = (u_1, u_2, u_3) \in \mathbb{R}^3$, si se especifican los cuaterniones $u = (0, u_1, u_2, u_3)$ y $Q = \cos\left(\frac{\theta}{2}\right) + \text{sen}\left(\frac{\theta}{2}\right)v$, siendo $\|Q\| = 1$, la expresión $Qu\bar{Q}$ define la resultante de rotar $\tilde{u}$ un ángulo de θ radianes alrededor de v en sentido positivo. En este caso, el cuaternión Q se denomina el cuaternión asociado a la rotación de un ángulo θ alrededor del vector unitario v , el cual coincide con el cuaternión asociado a una rotación de $-\theta$ radianes alrededor del vector unitario $-v$. Esta afirmación se deduce de la igualdad (88).

$$\cos\left(\frac{-\theta}{2}\right) + \text{sen}\left(\frac{-\theta}{2}\right)(-v) = \cos\left(\frac{\theta}{2}\right) + \text{sen}\left(\frac{\theta}{2}\right)v \quad (88)$$

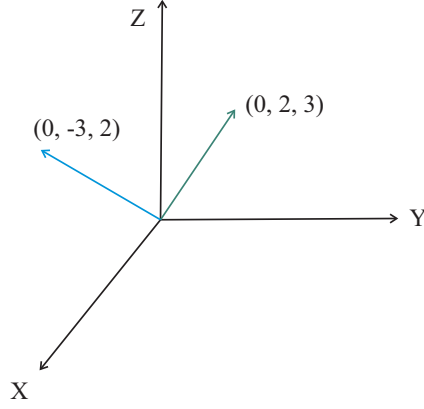
Es de anotar que cada una de las unidades i, j y k describe un giro de π radianes alrededor de los ejes X, Y o Z , respectivamente. El cuadrado de cada una de estas representa una rotación de 360 grados, es decir que tanto 1 como -1 señalan el mismo cambio, pero en sentidos opuestos.

Ejemplo 3

Al calcular la resultante de rotar el vector $\tilde{u} = (0, 2, 3)$ alrededor del eje X un ángulo de $\frac{\pi}{2}$ radianes, con el vector $v = 1$, $Q = \cos\frac{\pi}{4} + \text{sen}\frac{\pi}{4}i = \frac{\sqrt{2}}{2} + i\frac{\sqrt{2}}{2}$ y $u = (0, 0, 2, 3)$, se obtiene la expresión (89).

$$Qu\bar{Q} = (1-0)(0, 2, 3) \cdot (1, 0, 0)(1, 0, 0) + 0u + (1, 0, 0) \times (0, 2, 3) = (0, 0, -3, 2) \quad (89)$$

Así, el vector que se busca es $(0, -3, 2)$, el cual se ilustra en la figura 11.

Figura 11. Ejemplo de rotación en $\mathbb{R}^3$ por medio de cuaterniones**Fuente:** elaboración propia.

La expresión $Qu\bar{Q} = (1 - \cos\theta)(\tilde{u} \cdot v)v + \cos\theta\tilde{u} + \sin(\theta)v \times \tilde{u}$ se puede interpretar en forma matricial. Si se considera $v = (v_1, v_2, v_3)^T$ y $\tilde{u} = (u_1, u_2, u_3)^T$, el producto vectorial (90) se puede expresar matricialmente como se muestra en la expresión (91).

$$v \times \tilde{u} = (v_2u_3 - v_3u_2, v_3u_1 - v_1u_3, v_1u_2 - v_2u_1) \quad (90)$$

$$v \times \tilde{u} = \begin{pmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} \quad (91)$$

Por otra parte, al utilizar las expresiones (92) y (93), se obtiene la expresión (94).

$$\begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} (v_1 \quad v_2 \quad v_3) = \begin{pmatrix} v_1^2 & v_1v_2 & v_1v_3 \\ v_1v_2 & v_2^2 & v_2v_3 \\ v_1v_3 & v_2v_3 & v_3^2 \end{pmatrix} \quad (92)$$

$$\begin{aligned}
 (\tilde{u} \cdot v)v &= \begin{pmatrix} u_1 v_1^2 + u_2 v_1 v_2 + u_3 v_1 v_3 \\ u_1 v_1 v_2 + u_2 v_2^2 + u_3 v_2 v_3 \\ u_1 v_1 v_3 + u_2 v_2 v_3 + u_3 v_3^2 \end{pmatrix} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} (v_1 \quad v_2 \quad v_3) \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} \\
 &= \begin{pmatrix} v_1^2 & v_1 v_2 & v_1 v_3 \\ v_1 v_2 & v_2^2 & v_2 v_3 \\ v_1 v_3 & v_2 v_3 & v_3^2 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix}
 \end{aligned} \tag{93}$$

$$\begin{aligned}
 Qu\bar{Q} &= (1 - \cos\theta)(\tilde{u} \cdot v)v + \cos\theta \tilde{u} + \sin(\theta) v \times \tilde{u} \\
 &= (1 - \cos\theta)vv^T \tilde{u} + \cos\theta I \tilde{u} + \sin(\theta) \begin{pmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{pmatrix} \tilde{u}
 \end{aligned} \tag{94}$$

Esta última se puede representar por medio de la expresión (95).

$$Qu\bar{Q} = [(1 - \cos\theta)vv^T + \cos\theta I + \sin(\theta) \begin{pmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{pmatrix}] \tilde{u} \tag{95}$$

La expresión (96) representa la matriz de rotación, denotada por R_θ . Dicha matriz es ortogonal y v resulta ser un autovector de esta.

$$(1 - \cos\theta)vv^T + \cos\theta I + \sin\theta \begin{pmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{pmatrix} \tag{96}$$

Al desarrollar R_θ en forma matricial, se obtiene la expresión (97).

$$R_\theta = (1 - \cos\theta) \begin{pmatrix} v_1^2 & v_1 v_2 & v_1 v_3 \\ v_1 v_2 & v_2^2 & v_2 v_3 \\ v_1 v_3 & v_2 v_3 & v_3^2 \end{pmatrix} + \cos\theta \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} + \sin(\theta) \begin{pmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{pmatrix} \tag{97}$$

Al sustituir $\text{sen}(\theta) = S$ y $\cos \theta = C$ en la expresión anterior, esta se puede reescribir como se muestra en las expresiones (98) y (99).

$$R_\theta = \begin{pmatrix} (1-\cos\theta)v_1^2 + \cos\theta & (1-\cos\theta)v_1v_2 - \text{sen}(\theta)v_3 & (1-\cos\theta)v_1v_3 + \text{sen}(\theta)v_2 \\ (1-\cos\theta)v_1v_2 + \text{sen}(\theta)v_3 & (1-\cos\theta)v_2^2 + \cos\theta & (1-\cos\theta)v_2v_3 - \text{sen}(\theta)v_1 \\ (1-\cos\theta)v_1v_3 - \text{sen}(\theta)v_2 & (1-\cos\theta)v_2v_3 + \text{sen}(\theta)v_1 & (1-\cos\theta)v_3^2 + \cos\theta \end{pmatrix} \quad (98)$$

$$R_\theta = \begin{pmatrix} (1-C)v_1^2 + C & (1-C)v_1v_2 - Sv_3 & (1-C)v_1v_3 + Sv_2 \\ (1-C)v_1v_2 + Sv_3 & (1-C)v_2^2 + C & (1-C)v_2v_3 - Sv_1 \\ (1-C)v_1v_3 - Sv_2 & (1-C)v_2v_3 + Sv_1 & (1-C)v_3^2 + C \end{pmatrix} \quad (99)$$

Lo anterior indica que toda rotación en $\mathbb{R}^3$ se puede representar mediante la matriz R_θ .

Las matrices que indican rotaciones de θ radianes alrededor de alguno de los ejes coordenados x , y o z , se pueden deducir a partir de la matriz de rotación R_q que se ha obtenido. Al considerar los vectores $v_x=(1,0,0)$, $v_y=(0,1,0)$ y $v_z=(0,0,1)$, se generan las matrices que se muestran en (100), respectivamente.

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\text{sen}\theta \\ 0 & \text{sen}\theta & \cos\theta \end{pmatrix} \quad B = \begin{pmatrix} \cos\theta & 0 & \text{sen}\theta \\ 0 & 1 & 0 \\ -\text{sen}\theta & 0 & \cos\theta \end{pmatrix} \quad C = \begin{pmatrix} \cos\theta & -\text{sen}\theta & 0 \\ \text{sen}\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (100)$$

Estas matrices resultan ser ortogonales. Las matrices A , B y C se pueden interpretar como rotaciones en los planos YZ alrededor del eje X , XZ alrededor del eje Y y XY alrededor del eje Z , respectivamente.

Retomando el ejemplo anterior, al utilizar la matriz A , se tiene que el vector resultante se consigue planteando la operación (101).

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\text{sen}\theta \\ 0 & \text{sen}\theta & \cos\theta \end{pmatrix} \begin{pmatrix} 0 \\ 2 \\ 3 \end{pmatrix} \quad (101)$$

Si $\theta = \frac{\pi}{2}$, se obtiene la expresión (102).

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 2 \\ 3 \end{pmatrix} = \begin{pmatrix} 0 \\ -3 \\ 2 \end{pmatrix} \quad (102)$$

Este resultado coincide con el que se encontró mediante el tratamiento de cuaterniones. Sin embargo, este fue más sencillo de realizar.

Ahora, dado un cuaternión unitario $Q = q_1 + q_2i + q_3j + q_4k$, la rotación que determina este cuaternión se puede obtener a partir de la expresión $Qu\bar{Q}$. En este caso, las componentes del cuaternión unitario son las que se presentan en (103).

$$q_1 = \cos\left(\frac{\theta}{2}\right) \quad q_2 = v_1 \sin\left(\frac{\theta}{2}\right) \quad q_3 = v_2 \sin\left(\frac{\theta}{2}\right) \quad q_4 = v_3 \sin\left(\frac{\theta}{2}\right) \quad (103)$$

Siendo $v = (v_1, v_2, v_3)$ un vector unitario sobre el eje de rotación y $q \in [0, 2\pi)$ el ángulo de rotación, el cuaternión asociado a la rotación de un vector u está dado por la expresión $Qu\bar{Q}$. El propósito es obtener la representación matricial de la transformación en términos de las componentes del cuaternión Q . Primero, se parte de la matriz (104).

$$R_\theta = \begin{pmatrix} (1-C)v_1^2 + C & (1-C)v_1v_2 - Sv_3 & (1-C)v_1v_3 + Sv_2 \\ (1-C)v_1v_2 + Sv_3 & (1-C)v_2^2 + C & (1-C)v_2v_3 - Sv_1 \\ (1-C)v_1v_3 - Sv_2 & (1-C)v_2v_3 + Sv_1 & (1-C)v_3^2 + C \end{pmatrix} \quad (104)$$

Luego, se utilizan las expresiones (105).

$$\begin{aligned}
 1 - \cos\theta &= 2\operatorname{sen}^2\left(\frac{\theta}{2}\right) \\
 \cos\theta &= 1 - 2\operatorname{sen}^2\left(\frac{\theta}{2}\right) \\
 \operatorname{sen}\theta &= 2\operatorname{sen}\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right) \\
 \cos\theta &= \cos^2\left(\frac{\theta}{2}\right) - \operatorname{sen}^2\left(\frac{\theta}{2}\right)
 \end{aligned} \tag{105}$$

Ahora, se puede reescribir R_θ como se muestra en (106).

$$R_\theta = \begin{pmatrix} \cos^2\left(\frac{\theta}{2}\right) - \operatorname{sen}^2\left(\frac{\theta}{2}\right) + 2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_1^2 & 2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_1v_2 - 2\operatorname{sen}\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right)v_3 & 2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_1v_3 + 2\operatorname{sen}\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right)v_2 \\ 2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_1v_2 + 2\operatorname{sen}\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right)v_3 & 2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_2^2 + \cos^2\left(\frac{\theta}{2}\right) - \operatorname{sen}^2\left(\frac{\theta}{2}\right) & 2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_2v_3 - 2\operatorname{sen}\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right)v_1 \\ 2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_1v_3 - 2\operatorname{sen}\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right)v_2 & 2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_2v_3 + 2\operatorname{sen}\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right)v_1 & 2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_3^2 + \cos^2\left(\frac{\theta}{2}\right) - \operatorname{sen}^2\left(\frac{\theta}{2}\right) \end{pmatrix} \tag{106}$$

Al desarrollar las entradas de la diagonal principal de R_θ y utilizar el hecho de que $v_1^2 + v_2^2 + v_3^2 = 1$, se llega al resultado de la expresión (107).

$$\begin{aligned}
 \cos^2\left(\frac{\theta}{2}\right) - \operatorname{sen}^2\left(\frac{\theta}{2}\right) + 2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_1^2 &= \cos^2\left(\frac{\theta}{2}\right) - \operatorname{sen}^2\left(\frac{\theta}{2}\right)(1 - 2v_1^2) \\
 &= \cos^2\left(\frac{\theta}{2}\right) - \operatorname{sen}^2\left(\frac{\theta}{2}\right)(v_1^2 + v_2^2 + v_3^2 - 2v_1^2)
 \end{aligned} \tag{107}$$

A partir de esta, se obtiene la expresión (108).

$$\cos^2\left(\frac{\theta}{2}\right) - \operatorname{sen}^2\left(\frac{\theta}{2}\right) + 2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_1^2 = \cos^2\left(\frac{\theta}{2}\right) + \operatorname{sen}^2\left(\frac{\theta}{2}\right)(v_1^2 - v_2^2 - v_3^2) \tag{108}$$

De manera similar, se deduce la expresión (109).

$$\begin{aligned}
 2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_2^2 + \cos^2\left(\frac{\theta}{2}\right) - \operatorname{sen}^2\left(\frac{\theta}{2}\right) &= \cos^2\left(\frac{\theta}{2}\right) - \operatorname{sen}^2\left(\frac{\theta}{2}\right)(1 - 2v_2^2) \\
 &= \cos^2\left(\frac{\theta}{2}\right) - \operatorname{sen}^2\left(\frac{\theta}{2}\right)(v_1^2 + v_2^2 + v_3^2 - 2v_2^2)
 \end{aligned} \tag{109}$$

A partir de esta, se llega a la expresión (110).

$$\begin{aligned}
 2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_2^2 + \cos^2\left(\frac{\theta}{2}\right) - \operatorname{sen}^2\left(\frac{\theta}{2}\right) &= \cos^2\left(\frac{\theta}{2}\right) - \operatorname{sen}^2\left(\frac{\theta}{2}\right)(v_1^2 - v_2^2 + v_3^2) \\
 &= \cos^2\left(\frac{\theta}{2}\right) + \operatorname{sen}^2\left(\frac{\theta}{2}\right)(-v_1^2 + v_2^2 - v_3^2)
 \end{aligned} \tag{110}$$

De la misma forma, se obtiene la expresión (111).

$$2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_3^2 + \cos^2\left(\frac{\theta}{2}\right) - \operatorname{sen}^2\left(\frac{\theta}{2}\right) = \cos^2\left(\frac{\theta}{2}\right) - \operatorname{sen}^2\left(\frac{\theta}{2}\right)(v_1^2 + v_2^2 - v_3^2) \tag{111}$$

Al reemplazar (110) y (111), R_θ se convierte en (112).

$$\begin{aligned}
 R_\theta &= \\
 &\begin{pmatrix} \cos^2\left(\frac{\theta}{2}\right) + \operatorname{sen}^2\left(\frac{\theta}{2}\right)(v_1^2 - v_2^2 - v_3^2) & 2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_1v_2 - 2\operatorname{sen}\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right)v_3 & 2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_1v_3 + 2\operatorname{sen}\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right)v_2 \\ 2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_1v_2 + 2\operatorname{sen}\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right)v_3 & \cos^2\left(\frac{\theta}{2}\right) + \operatorname{sen}^2\left(\frac{\theta}{2}\right)(-v_1^2 + v_2^2 - v_3^2) & 2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_2v_3 - 2\operatorname{sen}\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right)v_1 \\ 2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_1v_3 - 2\operatorname{sen}\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right)v_2 & 2\operatorname{sen}^2\left(\frac{\theta}{2}\right)v_2v_3 + 2\operatorname{sen}\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right)v_1 & \cos^2\left(\frac{\theta}{2}\right) - \operatorname{sen}^2\left(\frac{\theta}{2}\right)(-v_1^2 - v_2^2 - v_3^2) \end{pmatrix}
 \end{aligned} \tag{112}$$

Las igualdades dadas en (113) se sustituyen en (112).

$$q_1 = \cos\left(\frac{\theta}{2}\right) \quad q_2 = v_1\operatorname{sen}\left(\frac{\theta}{2}\right) \quad q_3 = v_2\operatorname{sen}\left(\frac{\theta}{2}\right) \quad q_4 = v_3\operatorname{sen}\left(\frac{\theta}{2}\right) \tag{113}$$

La matriz R_θ se reescribe como se muestra en (114).

$$R_\theta = \begin{pmatrix} q_1^2 + q_2^2 - q_3^2 - q_4^2 & 2q_2q_3 - 2q_4q_1 & 2q_2q_4 + 2q_3q_1 \\ 2q_2q_3 + 2q_1q_4 & q_1^2 - q_2^2 + q_3^2 - q_4^2 & 2q_3q_4 - 2q_2q_3 \\ 2q_2q_4 - 2q_3q_1 & 2q_3q_4 + 2q_2q_1 & q_1^2 - q_2^2 - q_3^2 + q_4^2 \end{pmatrix} \quad (114)$$

Esta matriz indica la rotación que genera el cuaternión unitario $Q = q_1 + q_2i + q_3j + q_4k$. Del análisis anterior, se deduce que los cuaterniones sirven para representar rotaciones; sin embargo, no se pueden utilizar para definir traslaciones (Cova, 2019).

Descomposición simpléctica de un cuaternión

Sea $P = p_1 + p_2i + p_3j + p_4k$ un cuaternión en H , con $a = p_1 + p_2$ y $b = p_3 + p_4i$, se tiene que el cuaternión P se puede representar por medio de la expresión (115).

$$P = a + bj \quad (115)$$

Si se utiliza la idea de un operador complejo generalizado μ , es posible extrapolar la forma de representar a P .

Dados dos cuaterniones puros unitarios ortogonales entre sí, denotados como μ_1 y μ_2 , el cuaternión P se puede representar por medio de la expresión (116).

$$P = a + b\mu_2 \quad (116)$$

Siendo $a = p_1' + p_2'\mu_1$ y $b = p_3' + p_4'\mu_1$, en esta notación, a y b se conocen como la parte simple y la parte perpleja del cuaternión P , respectivamente.

Entonces, este se puede reescribir como se muestra en las expresiones (117) y (118).

$$P = (p_1' + p_2'\mu_1) + (p_3' + p_4'\mu_1)\mu_2 \quad (117)$$

$$P = p_1' + p_2'\mu_1 + p_3'\mu_2 + p_4'\mu_1\mu_2 = p_1' + p_2'\mu_1 + p_3'\mu_2 + p_4'\mu_3 \quad (118)$$

Dado que $\mu_3 = \mu_1\mu_2$ y μ_3 es ortogonal tanto a μ_1 como a μ_2 ($\mu_3 \perp \mu_1$) y ($\mu_3 \perp \mu_2$), se tiene un sistema de operadores isomorfo a los números cuaterniones i, j y k , el cual no está restringido a ser los tres ejes coordenados del espacio tridimensional.

El cuaternión $P = p_1 + p_2i + p_3j + p_4k$ en este sistema de operadores está dado por la expresión (119).

$$P = p_1' + p_2'\mu_1 + p_3'\mu_2 + p_4'\mu_3 \quad (119)$$

Además, se puede obtener utilizando la forma escalar y vectorial de P , como se muestra en (120).

$$\mu_3 = \mu_1\mu_2, P = s(P) + v(P); \quad s(P) = p_1'; \quad v(P) = p_2'\mu_1 + p_3'\mu_2 + p_4'\mu_3 \quad (120)$$

p_1', p_2', p_3' y p_4' están dadas en términos de las expresiones (121).

$$\begin{aligned} p_1' &= p_1 \\ -\frac{1}{2}(v(P)\mu_1 + \mu_1v(P)) &= v(P) \cdot \mu_1 = (p_2'\mu_1 + p_3'\mu_2 + p_4'\mu_3) \cdot \mu_1 = p_2' \\ -\frac{1}{2}(v(P)\mu_2 + \mu_2v(P)) &= v(P) \cdot \mu_2 = (p_2'\mu_1 + p_3'\mu_2 + p_4'\mu_3) \cdot \mu_2 = p_3' \\ -\frac{1}{2}(v(P)\mu_1\mu_2 + \mu_1\mu_2v(P)) &= v(P) \cdot \mu_1\mu_2 = (p_2'\mu_1 + p_3'\mu_2 + p_4'\mu_3) \cdot \mu_1\mu_2 = p_4' \end{aligned} \quad (121)$$

En resumen, se obtiene la expresión para el cuaternión $P = a + b \mu_2$ con $a = p_1' + p_2' \mu_1$ y $b = p_3' + p_4' \mu_1$, la cual se denomina descomposición simpléctica del cuaternión P . Para complementar este apartado, se puede consultar a Ell y Sangwine (2015).

Con la forma de descomposición simpléctica se consigue expresar un cuaternión en términos de dos planos perpendiculares, los cuales se interceptan solo en el origen en el espacio cuatridimensional y cada uno de estos puede ser interpretado como un plano complejo. Uno de los planos es el de la parte simple y tiene un eje real idéntico al eje escalar del espacio de cuaterniones; mientras que el otro plano corresponde a la parte perpleja del cuaternión y es perpendicular a ambos ejes del plano de la parte simple, su eje real es μ_2 y su eje imaginario es μ_3 .

Al utilizar $\mu_1 = \frac{i+j+k}{\sqrt{3}}$, se tiene que, al dividir una imagen a color en componentes paralelos y perpendiculares al eje, esto equivale a descomponer la imagen en sus factores de luminancia y crominancia. La parte simple será puramente imaginaria, ya que la luminancia requiere solo de un número real y la crominancia corresponderá a la parte perpleja, ya que esta requiere de dos números reales. Por tanto, al descomponer una imagen usando cuaterniones en forma simpléctica, se obtiene un par de imágenes complejas que pueden ser almacenadas de forma intercalada usando el mismo espacio como la imagen cuaternión. Esto quiere decir que cada valor de la forma $(0, R, G, B)$ corresponde al número cuaternión $a + b \mu_2$, siendo a estrictamente imaginario puro. Dichos valores se almacenan de manera independiente: por un lado, la parte real a y, por otro, la parte perpleja b , formando lo que se denomina en este contexto imagen cuaternión.

Algunas referencias útiles para ampliar la fundamentación teórica son Bergamini y Kamlofsky (2015) y Somma (2018).

Transformada discreta de Fourier de cuaterniones

Debido a la propiedad de no conmutatividad de los números cuaterniones, se desprenden tres tipos de transformada discreta de Fourier cuaternión (DQFT por sus siglas en inglés). Dependiendo del orden en que se efectúe la multiplicación cuaternión, esta se define con base en los conceptos de

multiplicación de cuaterniones, exponencial cuaternión y la propiedad de no conmutatividad de la multiplicación de cuaterniones. En este sentido, se hace referencia a la transformada de Fourier tipo 1, tipo 2 y tipo 3. Para conocer más del tema, se puede consultar a Ell y Sangwine (2015) y Said *et al.* (2008).

Dado un arreglo rectangular de números cuaterniones $f(m, n)$, la transformada de Fourier cuaternión bilateral (tipo 1) se define en la expresión (122).

$$F_{q_1}(p, s) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} e^{-\mu 2\pi \left(\frac{pm}{M}\right)} f(m, n) e^{-\mu 2\pi \left(\frac{sn}{N}\right)} \quad (122)$$

De manera similar, se definen las transformadas de Fourier unilateral izquierda y derecha en las expresiones (123) y (124), respectivamente.

$$F_{q_2}(p, s) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} e^{-\mu 2\pi \left(\frac{pm}{M} + \frac{sn}{N}\right)} f(m, n) \quad (123)$$

$$F_{q_3}(p, s) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) e^{-\mu 2\pi \left(\frac{pm}{M} + \frac{sn}{N}\right)} \quad (124)$$

De forma análoga, se definen las transformadas inversas discretas de Fourier cuaternión bilateral, izquierda y derecha como se muestra en las expresiones (125), (126) y (127), respectivamente.

$$f_{q_1}(m, n) = \sum_{p=0}^{M-1} \sum_{s=0}^{N-1} e^{\mu 2\pi \left(\frac{pm}{M}\right)} F_{q_1}(p, s) e^{-\mu 2\pi \left(\frac{sn}{N}\right)} \quad (125)$$

$$f_{q_2}(m, n) = \sum_{p=0}^{M-1} \sum_{s=0}^{N-1} e^{\mu 2\pi \left(\frac{pm}{M} + \frac{sn}{N}\right)} F_{q_2}(p, s) \quad (126)$$

$$f_{q_3}(m, n) = \sum_{p=0}^{M-1} \sum_{s=0}^{N-1} F_{q_3}(p, s) e^{\mu 2\pi \left(\frac{pm}{M} + \frac{sn}{N} \right)} \quad (127)$$

Estas también se conocen como transformadas tipo 1, tipo 2 y tipo 3, respectivamente. En esta definición, μ es un cuaternión unitario puro.

Cuando se hace referencia a los píxeles de una imagen o a la voz humana, a partir de muestreo se puede generar una secuencia finita. Si $\mu = i$, dichos datos se pueden concebir como valores de la forma $a = ib$. En ese caso, la transformada discreta de Fourier cuaternión se reduce a la transformada compleja de Fourier.

Una diferencia significativa entre la transformada discreta de Fourier cuaternión y la transformada discreta de Fourier compleja estándar es que la primera existe en tres versiones, dependiendo del orden del producto que aparezca en la expresión de la transformada.

La elección de μ , conocido como eje de la transformada, es arbitraria. En el caso de aplicaciones alusivas a imágenes en escala RGB, por razones de simetría, μ corresponde al eje gris en el espacio RGB, el cual conecta todos los puntos en los que $R = G = B$. Un valor de μ que se acostumbra a utilizar por conveniencia es $\frac{i+j+k}{\sqrt{3}}$.

La DQFT se emplea en el procesamiento de imágenes a color y la compresión de datos. Su implementación se ha facilitado un poco mediante el uso de algoritmos estándar de la transformada rápida de Fourier aplicados en aritmética de cuaterniones. Una forma es descomponer la transformada discreta de Fourier cuaternión en dos transformadas de Fourier complejas, lo cual es posible haciendo uso de la descomposición simpléctica de un cuaternión.

Partiendo del hecho de que un cuaternión arbitrario se puede expresar en dos componentes ortogonales: un eje arbitrario paralelo y un eje perpendicular a este, la transformada discreta de Fourier cuaternión se puede dividir en componentes paralelas y perpendiculares.

Si se considera que los ejes de transformación y descomposición de la imagen son el mismo y se utiliza un único eje perpendicular en ambos casos, se observa que la transformada discreta de Fourier cuaternión puede implementarse por medio de dos transformadas discretas de Fourier complejas, lo cual se describe a continuación.

Se tiene una secuencia de valores de una imagen $f(m, n)$ a partir de la expresión (128).

$$F_{q_2}(p, s) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} e^{-\mu_1 2\pi \left(\frac{pm}{M} + \frac{sn}{N} \right)} f(m, n) \quad (128)$$

La imagen a ser transformada en forma simpléctica se representa por medio de la expresión (129), donde $c_1(m, n) = w'(m, n) + \mu_1 x'(m, n)$ y $c_2(m, n) = y'(m, n) + \mu_1 z'(m, n)$.

$$f(m, n) = c_1(m, n) + c_2(m, n)\mu_2 \quad (129)$$

Cabe aclarar que, para el caso de imágenes RGB, se escoge $w'(m, n) = 0$. Usando el mismo cuaternión μ_1 para el eje de transformación, se puede reescribir la expresión de la transformada discreta de Fourier cuaternión tipo 2 como se muestra en la expresión (130), donde $\mu_1 \perp \mu_2$.

$$F_{q_2}(p, s) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} e^{-\mu_1 2\pi \left(\frac{pm}{M} + \frac{sn}{N} \right)} \left(c_1(m, n) + c_2(m, n)\mu_2 \right) \quad (130)$$

De la expresión (130), se obtienen las expresiones (131) y (132).

$$\begin{aligned}
F_{q_2}(p, s) &= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} e^{-\mu_1 2\pi \left(\frac{pm}{M} + \frac{sn}{N} \right)} c_1(m, n) \\
&\quad + \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} e^{-\mu_1 2\pi \left(\frac{pm}{M} + \frac{sn}{N} \right)} c_2(m, n) \mu_2 \\
&= C_1(p, s) + C_2(p, s) \mu_2
\end{aligned} \tag{131}$$

$$C_i(p, s) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} e^{-\mu_1 2\pi \left(\frac{pm}{M} + \frac{sn}{N} \right)} c_i(m, n); i = 1, 2 \tag{132}$$

La expresión $F_{q_2}(p, s)$ se conoce como descomposición simpléctica de la transformada discreta de Fourier cuaternión y resulta isomorfa a la transformada discreta de Fourier estándar compleja en dos dimensiones. Esto indica que la transformada discreta de Fourier cuaternión se puede calcular haciendo uso de la descomposición simpléctica de esta, lo que conlleva a calcular dos transformadas discretas de Fourier en dos dimensiones.

Se resalta el hecho de que, en la descomposición simpléctica de un cuaternión, es importante la definición de los ejes μ_1 y μ_2 . A partir del eje de la transformada μ_1 , se puede generar μ_2 , haciendo una elección arbitraria de un tercer cuaternión puro β que no sea paralelo a μ_1 .

Como el producto cruz $\mu_1 \times \beta$ es perpendicular tanto a μ_1 como a β , la restricción de módulo 1 se obtiene por normalización del producto cruz, de tal forma que la expresión para μ_2 está dada por (133).

$$\mu_2 = \frac{\mu_1 \times \beta}{\|\mu_1 \times \beta\|} = \frac{(\mu_1 \beta - \beta \mu_1)}{\|(\mu_1 \beta - \beta \mu_1)\|} \tag{133}$$

Es posible elegir β tal que una de las tres componentes de μ_2 sea cero, con el fin de reducir el número de multiplicaciones necesarias para expresar el cuaternión en forma simpléctica. Una manera de hacer esto es seleccionar uno de los tres cuaterniones unitarios i , j o k para que sea β , considerar $\mu_1 = \frac{i+j+k}{\sqrt{3}}$ y exigir que μ_2 tenga componente cero en la dirección de. Al realizar el producto cruz correspondiente con $\beta = i$, se obtienen las expresiones (134) y (135).

$$\mu_1 \times i = \frac{1}{\sqrt{3}} j - \frac{1}{\sqrt{3}} k \quad (134)$$

$$\|\mu_1 \times i\| = \sqrt{\frac{2}{3}} \quad (135)$$

Por tanto, $\mu_2 = \frac{j-k}{\sqrt{2}}$.

Dado que $\mu_1 \times \mu_2 = \frac{1}{\sqrt{6}}(-2i + k + j)$, se deduce de manera general que μ_1 y μ_2 son ortogonales.

Del razonamiento anterior, se concluye que la transformada discreta de Fourier cuaternión se puede implementar a partir de la descomposición de dos transformadas isomórficas en la transformada discreta de Fourier compleja. Por tanto, se puede calcular a partir de la transformada rápida de Fourier FFTS.

Un modelo criptográfico basado en caos junto con la teoría de cuaterniones se describe en la sección Aplicación de la teoría de cuaterniones del capítulo 4, así como en el artículo de Mendez *et al.* (2023).

Referencias

Altmann, S. L. (1989). Hamilton, Rodrigues, and the Quaternion Scandal. *Mathematics Magazine*, 62(5), 291-308. <https://doi.org/10.2307/2689481>

- Bannon, T. (2015). The Origin of Quaternions. *The College Mathematics Journal*, 46(1), 43-50. <https://doi.org/10.4169/college.math.j.46.1.43>
- Bergamini, M. L. y Kamlofsky, J. A. (2015, del 19 al 20 de noviembre). Parametrizaciones de movimientos rígidos en 3D para visión artificial. *3er. Congreso Nacional de Ingeniería Informática / Sistemas de Información CoNaIISI*. Buenos Aires, Argentina. <https://imgbiblio.vaneduc.edu.ar/fulltext/files/TC119399.pdf>
- Cova, W. J. D. (2019). *Cuaterniones: visualización de rotaciones de cuerpos rígidos* (1.^a ed.). edUTecNe. https://ria.utn.edu.ar/bitstream/handle/20.500.12272/4486/CUATERNIONES_Visualizacion_Rotaciones_%28W_Cova%29.pdf?sequence=6&isAllowed=y
- Ell, T. A. y Sangwine, S. J. (2000). Decomposition of 2D Hypercomplex Fourier Transforms into Pairs of Complex Fourier Transforms. *2000 10th European Signal Processing Conference*, 2, 1061-1064. <https://www.eurasip.org/Proceedings/Eusipco/Eusipco2000/SESSIONS/WEDPM/SS2/CR1084.PDF>
- ElSharkawy, M. I. (2017). Integrating and Securing Video, Audio and Text Using Quaternion Fourier Transform. *International Journal of Communication Networks and Information Security (IJCNIS)*, 9(3), 412-419. <https://doi.org/10.17762/ijcnis.v9i3.2634>
- Kamlofsky, J. A. y Bergamini, M. L. (2015). Los cuaterniones en visión robótica. *Revista Matemática Aplicada, Computacional e Industrial (MACI)*, 5, 517-520.
- Kapoor, R., Verma, S. y Kapoor, A. (2021). Quaternions-based Detection of Power Quality Events. *Electrical Engineering*, 103, 2911-2927. <https://doi.org/10.1007/s00202-021-01263-z>
- Mahecha, V. A. (2008). Aplicación de los números hipercomplejos o cuaterniones en imágenes de color. *Ingeniería y Desarrollo*, (23), 72-83. <https://www.redalyc.org/articulo.oa?id=85202307>
- Mendez, L., Ladino, D., Amaya, I. y Alvarado, D. (2023). A New Multimedia Cryptosystem Using Chaos, Quaternion Theory and Modular Arithmetic. *Multimedia Tools and Applications*. <https://doi.org/10.1007/s11042-023-14475-1>
- Rubio, M. A. (2019). *Unicidad de los cuaterniones y octoniones como R-álgebras de división y sus representaciones matriciales* [Tesis de maestría, Universidad Nacional Mayor de San Marcos]. CYBERTESIS.

https://cybertesis.unmsm.edu.pe/bitstream/handle/20.500.12672/11640/Rubio_gm.pdf?sequence=3&isAllowed=y

Said, S., Le Bihan, N. y Sangwine, S. J. (2008). Fast Complexified Quaternion Fourier Transform. *IEEE Transactions on Signal Processing*, 56(4), 1522-1531. <https://doi.org/10.1109/TSP.2007.910477>

Sánchez Muñoz, J. M. (2011). Historias de matemáticas - Hamilton y el descubrimiento de los cuaterniones. *Pensamiento Matemático*, (1), 1-27. https://www2.caminos.upm.es/Departamentos/matematicas/revis-tapm/revista_impresa/numero_1/hamilton_y_el_descubrimiento_de_los_cuaterniones.pdf

Somma, F. J. (2018). *Cuaterniones y ángulos de Euler para describir rotaciones en $\mathbb{R}^3$* [Tesis de pregrado, Universidad Abierta Interamericana]. VANEDUC. <https://imgbiblio.vaneduc.edu.ar/fulltext/files/TC126683.pdf>

Capítulo 3. Autómatas celulares

Reseña histórica

Los autómatas celulares (AC) fueron planteados entre finales de 1940 e inicios de 1950 por Von Neumann (Aprendemos matemáticas, s. f.; Carvalho *et al.*, 2019; Kari, 2005b; Reyes Gómez, 2011; Sancho Caparrini, 2016), quien, inspirado en la biología, trataba de diseñar sistemas artificiales capaces de autorreplicarse. Neumann estaba interesado en modelar sistemas análogos al cerebro humano, donde la memoria y el procesamiento no están separados, son masivamente paralelos y tienen la capacidad de repararse a sí mismos. Entonces, planteó un modelo matemático con reglas que operaban sobre una malla rectangular, las cuales fueron interpretadas como un conjunto de células capaces de crecer, reproducirse y morir a través del tiempo. Por esto, han tomado cada vez más interés en el modelado de sistemas naturales, lo que implica que se pueden encasillar dentro del estudio de los sistemas complejos.

El trabajo de Neumann se basó en los aportes de Konrad Zuse y Stanisław Ulam. El ingeniero alemán Zuse concebía un computador como un modelo discreto de un sistema físico. A su vez, el matemático polaco

Stanisław Ulam consideró un arreglo rectangular de celdas como un tablero de ajedrez en el cual cada celda podía estar en uno de un número finito de estados, y el tiempo se desarrollaba a saltos discretos. Durante cada salto de tiempo, las celdas podían cambiar su estado. (Fernández Fraga y Rangel Mondragón, 2014, p. 13)

Alrededor de la década de 1960, los autómatas celulares se empezaron a estudiar como un sistema dinámico discreto que puede presentar un comportamiento caótico o no, según la función de transición que regule la regla de su evolución, motivo por el que John Holland comenzó a aplicarlos en problemas de optimización y adaptación. Durante la década de 1970, con base en la propuesta de Neumann, el matemático británico John Horton Conway propuso lo que en la actualidad se conoce como *juego de la vida*. Este consistía en una colección de celdas y unas reglas matemáticas que implicaban que las células podían vivir, morir o multiplicarse, dependiendo de las condiciones iniciales (Fernández Fraga y Rangel Mondragón, 2014).

En este capítulo se explicarán detalladamente las características y el comportamiento de los autómatas celulares tanto de una como de dos dimensiones. Sin embargo, el proceso podrá ser extrapolado a infinitas dimensiones.

Fundamentación teórica

Tal como se mencionó anteriormente, los AC pueden verse como *mallas* compuestas por células relacionadas entre sí, de acuerdo con la posición que ocupan. Las células cambian sus estados sincrónicamente dependiendo de los valores de algunas células cercanas (vecinas), según lo defina la regla de actualización local que se esté empleando. Usualmente, todas las células aplican la misma regla de actualización; de esta manera, el sistema es homogéneo. Para que el comportamiento de los AC se comprenda fácilmente, se analizarán únicamente autómatas sincrónicos en una cuadrícula rectangular infinita.

Formalización de autómatas celulares

De acuerdo con Cano Rojas y Rojas Matas (2016), Carvalho *et al.* (2019) y Kari (2005a), un autómata celular se puede representar por medio de una cuádrupla (L, S, n, f) , donde:

L es una región regular de dimensión d y los elementos de L se llaman celdas o células.

S representa el conjunto finito de estados o valores que pueden tomar las celdas.

n , corresponde al conjunto finito de vecindades de cada celda.

f es la función de transición o regla de transición $f: S^n \rightarrow S$, que describe la evolución del autómata a partir de una condición inicial llamada *configuración inicial del autómata*.

La dimensión hace énfasis en el espacio en el que evoluciona el AC a través del tiempo dentro de un área restringida. La notación $d = 1$ (ACD1) representa un autómata lineal unidimensional (vector); mientras que $d = 2$ (ACD2) indica un autómata bidimensional que se representa matricialmente.

El conjunto de estados de un AC corresponde a todos los valores que puede tomar una celda. El más sencillo de ellos está compuesto por dos valores (usualmente 0 y 1) denominados elementos biestables. Además, cada estado se puede representar por un vector de componentes enteros, reales o de valores simbólicos, por ejemplo: $S = \{0,1,2,3\}$ o $S = \{\text{amarillo, azul, rojo}\}$.

Para un conjunto de estados S , la configuración de un autómata, en un instante cualquiera, está dada por la función $c: \mathbb{Z}^d \rightarrow S$, la cual especifica el estado de todas las celdas, siendo $\mathbb{Z}^d$ un autómata celular de dimensión d , es decir, una posible configuración en un instante. Por su parte, el conjunto $S^{\mathbb{Z}^d}$ de todas las posibles configuraciones se denota como $c(d, S)$. El estado e_t siguiente de una célula depende del estado de sus vecinas en un momento anterior $t - 1$ y corresponde a la configuración obtenida después de aplicar la regla sobre la configuración c (Kari, 2005b).

Comportamiento dinámico de los autómatas celulares

De acuerdo con Cattaneo *et al.* (1999) y Romero (2012) y con base en las anteriores definiciones de autómata celular, es posible afirmar que un AC es un sistema dinámico que evoluciona en tiempos discretos, donde sus configuraciones se corresponden con sucesiones d-dimensionales y presentan un espacio de estados que posee un conjunto finito de valores posibles.

Las células que componen un AC cambian sus estados sincrónicamente en pasos de tiempo discretos, es decir, la regla es aplicada a todas ellas simultáneamente, en cada iteración. El siguiente estado de cada célula depende de los estados actuales de sus vecinas de acuerdo con la regla de actualización.

Según Kari (2005a), una manera formal de especificar la vecindad es la siguiente: “Sea $N = (\vec{x}_1, \vec{x}_2, \dots, \vec{x}_n)$ un vector de distintos elementos de $\mathbb{Z}^d$. Los vecinos de una célula en la ubicación $\vec{x} \in \mathbb{Z}^d$ son las n células de las ubicaciones $\vec{x} + \vec{x}_i$ para $i = 1, 2, 3, \dots, n$ ”, donde d corresponde a la dimensión y n al tamaño de la vecindad.

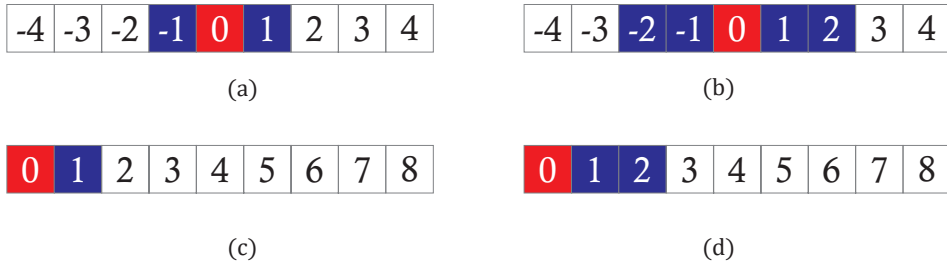
Por otra parte, la regla de transición es una función $f : S^n \rightarrow S$, cuyo valor $e_{i, t+1} = f \{e_{1t}, e_{2t}, \dots, e_{nt}\}$ en un tiempo $t + 1$ corresponde al nuevo estado de una célula i , cuyos n vecinos estaban en los estados $e_1, e_2, \dots, e_n$, un momento antes, es decir, en el tiempo t . La aplicación continua de dicha regla sobre cada una de las células determina el comportamiento dinámico global del autómata celular.

Tipos de vecindad

Las vecindades hacen referencia al conjunto de celdas próximas que ejercen influencia en el estado de una celda determinada; con frecuencia, esta proximidad es definida en términos de un radio r . Las vecindades más usadas son la vecindad de Von Neumann (NVN) y la vecindad de Moore (NM). En la NVN, se suman o se restan vectores con la i -ésima coordenada unitaria y_i a la posición de la celda, considerando que esta tiene $2r + 1$ vecinos incluyendo la misma celda. En la NM, se suma un vector, donde cada valor puede ser $-1, 0$ o 1 , a la posición de la celda, considerando que cada célula tiene $3r$ vecinos, incluyendo la misma celda (Fernández Fraga y Rangel Mondragón, 2014; Kumar y Kumar Sharma, 2022).

Por lo general, en la vecindad de Moore, se obtiene el radio r del autómata celular para cualquier entero r positivo. Para el caso de radio $\frac{1}{2}$ significa que la vecindad contiene compensaciones $y = (y_1, y_2, \dots, y_d)$, donde cada y_i es 0 o 1 . Un radio unidimensional $\frac{1}{2}$ de un AC también se denomina de una vía. Esta vecindad no contiene elementos negativos.

En la figura 12, se ilustran gráficamente algunos ejemplos de vecindades para diferentes radios y números de celdas vecinas en autómatas unidimensionales. En todos los casos, la celda de referencia es de color rojo, mientras que sus vecinas son de color azul.

Figura 12. Vecindad de dimensión 1

Fuente: Fernández Fraga y Rangel Mondragón (2014).

Los autómatas de la figura 12 son de una sola dimensión. Las especificaciones para la vecindad de la figura 12(a) son $d = 1$ y $N(1) = 2$, que significan dimensión: 1, distancia (radio): 1 y celdas vecinas: 2. La notación $N(1)$ hace referencia al conjunto de vecindades de una celda, es decir que se tiene una vecina a cada lado de la celda de referencia (0).

Para la figura 12(b), las especificaciones son $d = 1$ y $N(2) = 4$, que significan dimensión: 1, distancia (radio): 2 y celdas vecinas: 4, es decir que se tienen dos vecinas a cada lado de la celda de referencia.

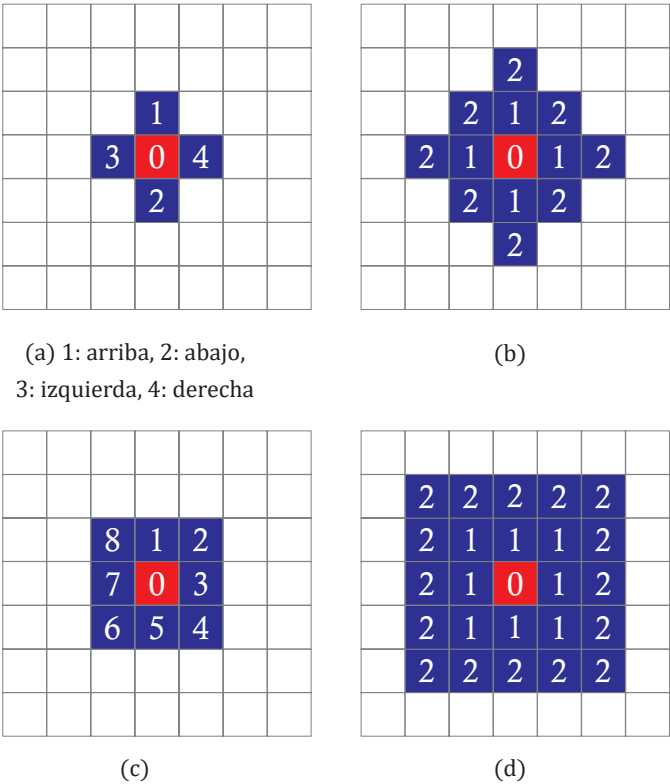
Para el autómata de la figura 12(c), las especificaciones son $d = 1$ y $N(1) = 1$, que significan dimensión: 1, distancia (radio): 1 y celdas vecinas: 1, es decir que la celda de referencia tiene solamente una vecina.

Finalmente, para el autómata de la figura 12(d), las especificaciones son $d = 1$ y $N(2) = 2$, que significan dimensión: 1, distancia (radio): 2 y celdas vecinas: 2, es decir que la celda de referencia tendrá dos celdas vecinas, pero solamente a uno de sus dos lados, ya que el radio es 2. Cuando se consideran únicamente celdas a un solo lado de la referencia, se suelen tomar las celdas del lado derecho.

Cuando se trabajan dos dimensiones, se pueden observar dos tipos de vecindad: la NVN y la NM. En la figura 13, las dos primeras imágenes (a) y (b) hacen referencia a la NVN y las dos últimas (c) y (d), a la NM. Cabe resaltar que la celda de referencia está identificada con color rojo y, para este caso, con el número cero, mientras que las vecinas se identifican con color azul.

En la figura 13(a), se observa un autómata con las especificaciones $d = 2$ y $N(1) = 4$, que significan dimensión: 2, distancia (radio): 1 y celdas vecinas: 4. Si la celda de referencia se ubica dentro del primer cuadrante en la posición (x, y) , las vecinas corresponden a aquellas ubicadas en las posiciones $(x, y - 1)$ arriba, $(x, y + 1)$ abajo, $(x - 1, y)$ a la izquierda y $(x + 1, y)$ a la derecha.

Figura 13. Vecindad de dimensión 2



Fuente: elaboración propia a partir de Fernández Fraga y Rangel Mondragón (2014).

Otro ejemplo de la NVN se presenta en la figura 13(b) con las especificaciones $d = 2$ y $N(2) = 12$, que significan dimensión: 2, distancia (radio): 2 y celdas vecinas: 12. En este caso, se consideran celdas vecinas todas las que rodean la de referencia y, adicionalmente, las que se ubican arriba, abajo, a la izquierda y a la derecha de la siguiente fila o columna, respecto a la base (celda roja). En esta imagen se pueden observar las celdas que hacen parte de lo que se considera radio 1 y aquellas que constituyen el radio 2.

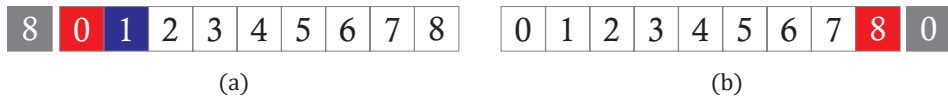
En la figura 13(c), se presenta la NM con las especificaciones $d = 2$ y $N(1) = 9$, que significan dimensión: 2, distancia (radio): 1 y celdas vecinas: 9. Como se puede observar, corresponde a todas las celdas alrededor de la referencia, incluyendo esta última.

Finalmente, la figura 13(d) también corresponde a la NM y tiene las especificaciones $d = 2$ y $N(2) = 25$, que significan dimensión: 2, distancia (radio): 2 y celdas vecinas: 25. Al igual que en el caso anterior, corresponde a todas las celdas alrededor de la referencia, incluyendo esta última, que pertenecen a las dos filas y columnas más cercanas. Las celdas que se consideran dentro del radio 1 se identifican con 1 y las que constituyen el radio 2, con 2.

Un caso especial se presenta cuando se quieren identificar las vecindades en los bordes, ya que dependen de las condiciones que se impongan en los límites de la región. Debido a esto, se pueden tener bordes periódicos, absorbentes, reflectores o abiertos (Fernández Fraga y Rangel Mondragón, 2014; Mohammed y Nayak, 2014; Vargas-Forero *et al.*, 2019).

Bordes periódicos: se caracterizan porque las celdas opuestas son consideradas vecinas (torus). En la figura 14 se muestran dos ejemplos de este tipo de bordes para autómatas de una dimensión.

Figura 14. Vecindad de bordes periódicos, dimensión 1



Fuente: elaboración propia a partir de Fernández Fraga y Rangel Mondragón (2014).

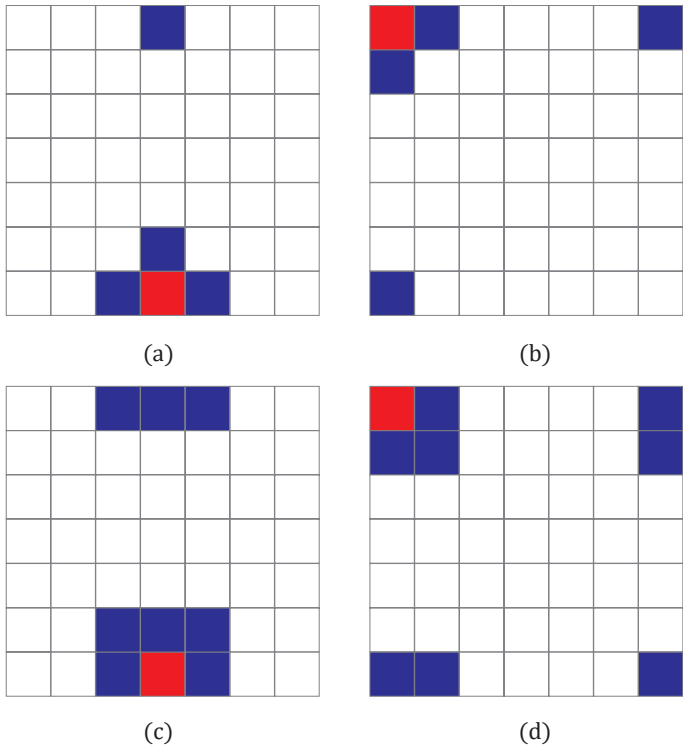
En el primer caso, se tienen las especificaciones $d = 1$ y $N(1) = 2$, que significan dimensión: 1, distancia (radio): 1 y celdas vecinas: 2. En el segundo caso, se tienen las especificaciones $d = 1$ y $N(1) = 1$, que significan dimensión: 1, radio: 1 y celdas vecinas: 1. Como se puede observar, el borde periódico hace que se considere como vecina la celda del otro extremo.

En la figura 15, se observa que los bordes periódicos para espacios de dos dimensiones se comportan de manera similar. Las figuras 15(a) y 15(b) hacen referencia a la vecindad de NVN con las especificaciones $d = 2$ y $N(1) = 4$, mientras que las figuras 12(c) y 12(d) aluden a la vecindad NM con

especificaciones $d = 2$ y $N(1) = 9$ y. En ambos casos, se evidencian bordes periódicos.

En este contexto, se evidencia que, para identificar las celdas vecinas, es necesario imaginar el plano como una figura de donut o de torus, donde la parte superior toca la inferior y la parte izquierda toca la derecha.

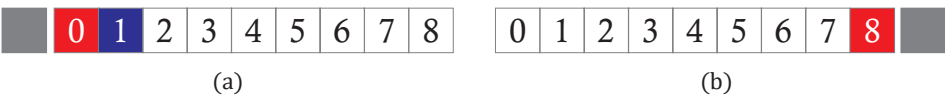
Figura 15. Vecindad de bordes periódicos, dimensión 2



Fuente: elaboración propia a partir de Fernández Fraga y Rangel Mondragón (2014).

Bordes absorbentes: este tipo de vecindad considera que las celdas de los bordes no tienen vecinos. En la figura 16, se presentan dos ejemplos para autómatas de una dimensión.

Figura 16. Vecindad de bordes absorbentes, dimensión 1



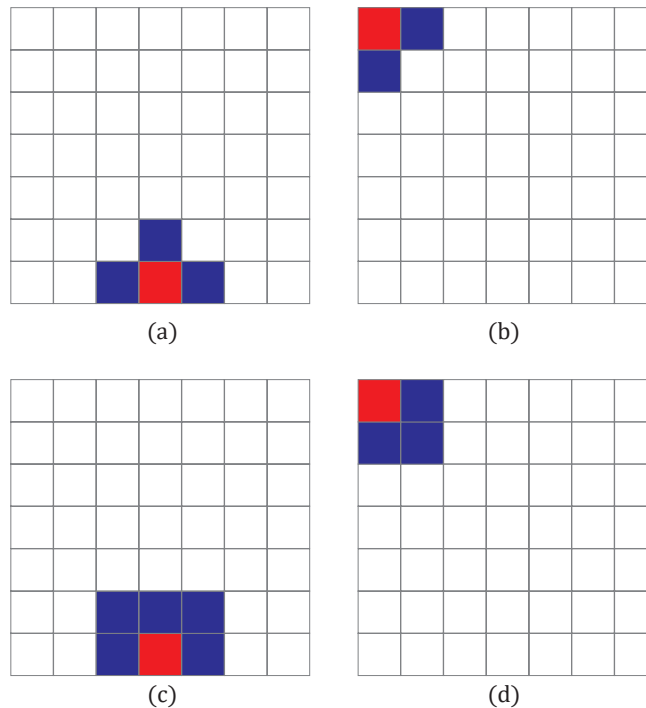
Fuente: elaboración propia a partir de Fernández Fraga y Rangel Mondragón (2014).

La figura 16(a) tiene las especificaciones $d = 1$ y $N(1) = 2$, es decir que tiene 2 celdas vecinas; sin embargo, debido a que se utilizan bordes absorbentes, solamente se tendrá en cuenta una celda vecina. En la figura 16(b), se muestra un autómata con las especificaciones $d = 1$ y $N(1) = 1$, es decir que tiene 1 celda vecina; sin embargo, el uso de bordes absorbentes hace que no se contemple ningún vecino.

De forma análoga, al aplicar bordes absorbentes a los autómatas de dos dimensiones, los vecinos externos no se tendrán en cuenta.

Las figuras 17(a) y 17(b) corresponden a la vecindad NVN con las especificaciones $d = 2$ y $N(1) = 4$. Las figuras 17(c) y 17(d) corresponden a la vecindad NM con las especificaciones $d = 2$ y $N(1) = 9$. En ambos casos, se usan bordes absorbentes.

Figura 17. Vecindad de bordes absorbentes, dimensión 2



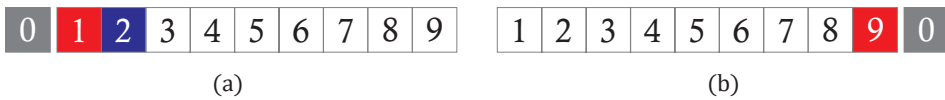
Fuente: Fernández Fraga y Rangel Mondragón (2014).

Bordes reflectores: también se denominan bordes adiabáticos. En este caso, las celdas del borde se replican considerando su mismo valor como vecino. En

La vecindad NVN con las especificaciones $d = 2$ y $N(1) = 4$ corresponde a las figuras 19(a) y 19(b). La vecindad NM con las especificaciones $d = 2$ y $N(1) = 9$ corresponde a las figuras 19(c) y 19(d). En todos los casos, se observa claramente que el valor de las celdas que se encuentran al borde del espacio se replica para considerarlo como vecino de la referencia.

Bordes abiertos: en ocasiones, se denominan bordes fijos porque se considera que todas las celdas externas tienen el mismo valor, el cual puede ser un valor lógico (1: verdadero o 0: falso) según el caso. Para los ejemplos que se muestran a continuación, se asume que el valor lógico de las celdas externas es falso, es decir que tienen asignado cero.

Figura 20. Vecindad de bordes abiertos, dimensión 1

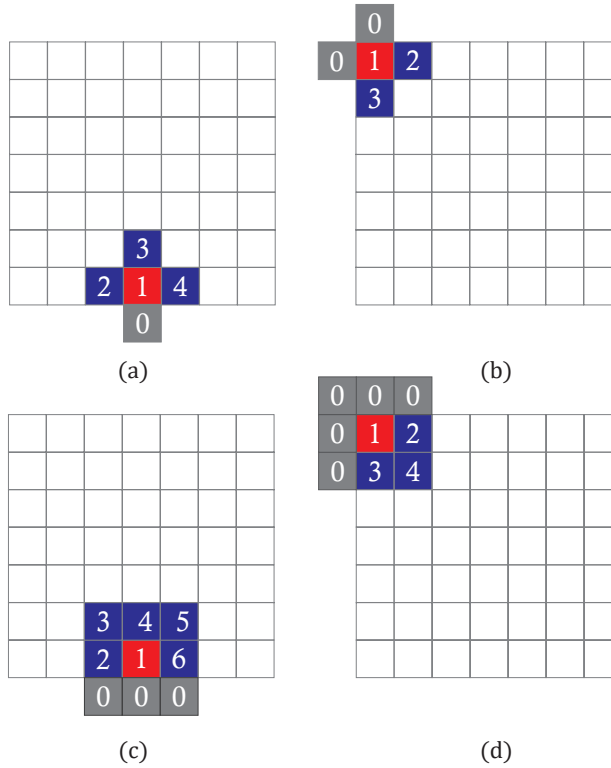


Fuente: elaboración propia a partir de Fernández Fraga y Rangel Mondragón (2014).

En la figura 20, se presentan dos ejemplos de autómatas de una dimensión con bordes abiertos. En la figura 20(a), las especificaciones son $d = 1$ y $N(1) = 2$, es decir que tiene 2 vecinos, uno a cada lado. Como se mencionó antes, se observa que la celda roja (con el número 1) tiene como vecino izquierdo (vecino de borde) el valor fijo predeterminado (0). En la figura 20(b), las especificaciones son $d = 1$ y $N(1) = 1$, que implica que la celda de referencia, ubicada en el borde del espacio, solo tiene un vecino que también toma el valor predeterminado (0).

La definición de los vecinos con bordes abiertos para espacios de dos dimensiones es análoga a la de una dimensión. Las figuras 21(a) y 21(b) presentan dos ejemplos para la NVN y las figuras 21(c) y 21(d) presentan dos ejemplos para la NM. La celda de referencia es de color rojo y se identifica con el número 1, las celdas vecinas son de color azul y se identifican con los números 2 y 3 para la NVM y con los números del 2 al 6 para la NM, y los bordes son de color gris y se identifican con el número 0.

Figura 21. Vecindad de bordes abiertos, dimensión 2



Fuente: elaboración propia a partir de Fernández Fraga y Rangel Mondragón (2014).

Características de los autómatas celulares

Los AC se caracterizan por ser masivamente paralelos y homogéneos, todas sus interacciones son locales y su proceso dinámico hace necesario establecer los valores iniciales de cada una de las celdas, los cuales en conjunto se denominan *condiciones iniciales* o *edén*. Otro elemento fundamental de los AC, además de las condiciones iniciales, es la *regla de transición* o *función de transición*, la cual determina el siguiente estado de cada celda. Existen varios tipos de reglas: determinísticas, totalísticas y totalísticas exteriores, entre otras (Fernández Fraga y Rangel Mondragón, 2014).

Reglas determinísticas

En este caso, se define una regla para asignar el valor de la celda en cada una de las combinaciones posibles de los estados. El número total de reglas

depende de la cantidad de celdas consideradas vecinas, así como del número de estados de cada celda, pues deben existir reglas para todas las posibles combinaciones tanto de los valores de las celdas vecinas como de los valores asignados para el siguiente estado de la celda. Supóngase que se consideran n celdas vecinas (usualmente, $n = 2r + 1$, donde r corresponde al radio de la vecindad) y que cada celda puede tener uno de K estados posibles; de esta forma, el número de combinaciones de celdas vecinas con todos los posibles estados en cada una de ellas es K^n . Si se tiene en cuenta que una regla asigna un solo valor a cada una de las combinaciones y que existen K valores posibles, el número total de reglas determinísticas que se obtienen es K^{K^n} . Por ejemplo, para el caso de las reglas de Wolfram (que se analizarán más adelante), donde $r = 1$ y cada celda es biestable, se tiene que $n = 3$, por tanto, el número de reglas es $2^{2^3} = 256$.

Reglas totalísticas

El estado actual de la celda depende de la suma de sus celdas vecinas, lo cual podría expresarse como $e_{t+1} = f(\sum N(r):e_t)$. Por ejemplo, si se tiene un autómata con las especificaciones $d = 1$, $N(1) = 2$ y $S = \{0,1\}$, es decir, de dimensión 1, con un conjunto de 2 vecindades de distancia 1 y un conjunto de dos posibles valores para cada celda (0 o 1), una regla totalística expresada como $e_{t+1} = f(<2 \rightarrow 0, 2 \rightarrow 1)$ significa que, si la suma de los vecinos es menor que 2, su nuevo estado es 0; mientras que, si la suma es igual a 2, su nuevo estado es 1.

Reglas totalísticas exteriores

Este tipo de reglas se comportan de manera similar a las totalísticas con la diferencia de que, en este caso, se tiene en cuenta el valor de la celda misma, además del de sus vecinas. Esto puede expresarse como $e_{t+1} = f(e_t, \sum(N(r):e_t))$. Para un autómata con las especificaciones $d = 1$, $N(1) = 2$ y $S = \{0,1\}$, una regla totalística exterior podría expresarse como $e_{t+1} = f((0,0) \rightarrow 0, (0,x > 0) \rightarrow 1, (1,x > 0) \rightarrow 0, (1,0) \rightarrow 1)$, lo cual implica que $e_{t+1} = 0$ si se cumple que $e_t = 0$ y la suma de sus vecinos también es igual a 0 o si $e_t = 1$ y la suma de sus vecinos es mayor que 0. Asimismo, $e_{t+1} = 1$ si se cumple que $e_t = 0$ y la suma de sus vecinos es mayor que 0 o si $e_t = 1$ y la suma de sus vecinos es igual a 0.

Al aplicar en repetidas ocasiones algunas de estas reglas sobre un edén dado y observar de manera global la evolución de todas las celdas en la región, se pueden descubrir patrones o comportamientos caóticos, que a su vez se enmarcan dentro de las siguientes tipologías:

- Las reglas o funciones de corrimiento se caracterizan porque trasladan la configuración una célula abajo en una de las direcciones de las coordenadas.
- En otras ocasiones, se mantiene el mismo estado cada vez que se aplica la regla. Esto se conoce como estado quiescente o estable, es decir que la configuración quiescente es aquella donde todas las células son estables.

Una configuración $c \in S^{\mathbb{Z}^d}$ se denomina finita si solo un número finito de células son no-quiescentes.

Una evolución periódica o más precisamente espacialmente periódica es aquella invariante bajo d traslaciones linealmente independientes. Las configuraciones finitas y periódicas se emplean en simulaciones efectivas de AC en computadores (Fernández Fraga y Rangel Mondragón, 2014).

Tipos de autómatas

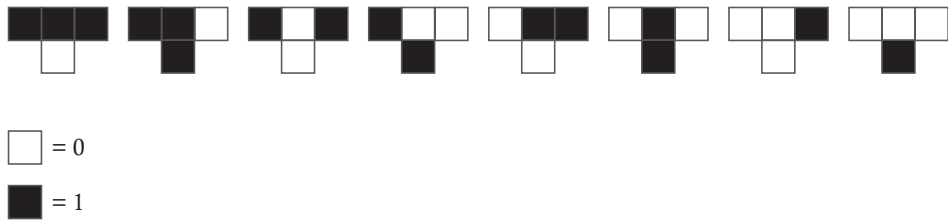
Los autómatas celulares pueden ser de una o más dimensiones de acuerdo con la región que se haya definido para aplicar la función de transición. A continuación, se revisará el comportamiento de los autómatas de una y dos dimensiones.

Autómatas unidimensionales o elementales

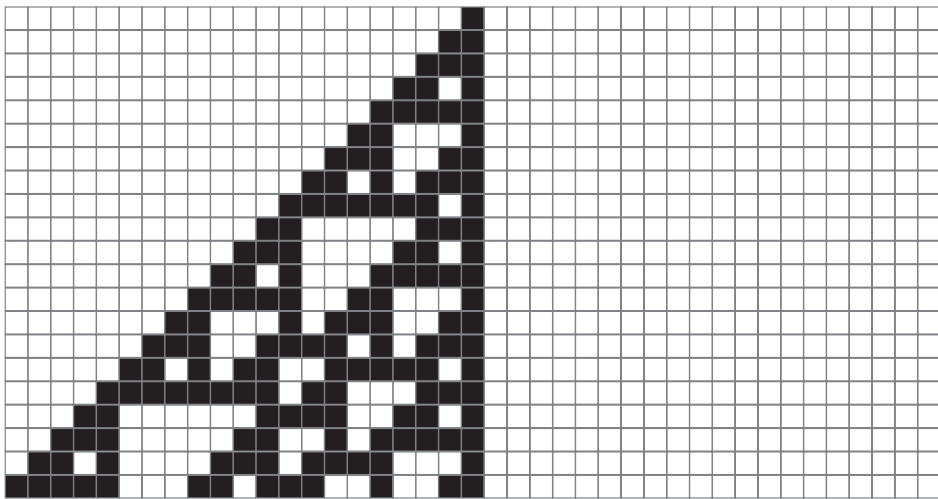
Hacia 1980, Stephen Wolfram realizó algunas publicaciones donde presentaba un tipo de autómatas celulares que denominó elementales. En estas describía explícitamente las reglas de evolución (usando reglas determinísticas) y las configuraciones obtenidas después de una iteración (Fernández Fraga y Rangel Mondragón, 2014; Kari, 2005a).

En la figura 22, se presenta la regla 110 de Wolfram y el resultado de algunas iteraciones de esta regla, partiendo de una celda central como edén.

Figura 22. Regla 110 de Wolfram



(a) Esquema de la regla 110



(b) Evolución de la regla 110, ejecutada en Wolfram Alpha

Fuente: elaboración propia.

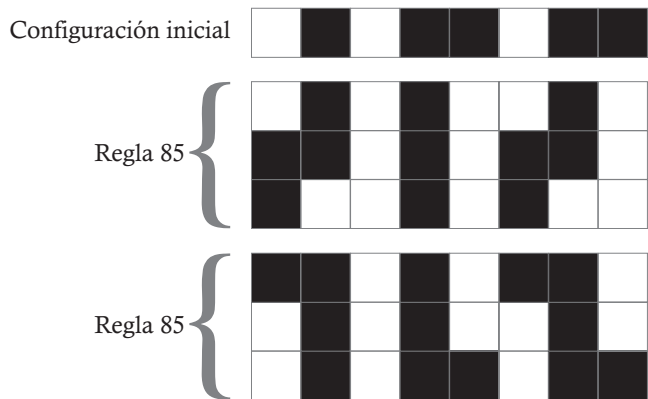
A partir de un análisis del comportamiento cualitativo de los autómatas elementales, Wolfram propuso la siguiente clasificación relacionada con la forma en la que evoluciona la función de transición:

- Clase 1, autómatas celulares de tipo fijo: los patrones iniciales evolucionan a un estado estable, es decir que, después de algunas iteraciones, se quedan en el mismo valor.
- Clase 2, autómatas celulares periódicos: la configuración inicial evoluciona a un estado donde se repite el mismo patrón.

- Clase 3, autómatas celulares caóticos: estos evolucionan generando secuencias pseudoaleatorias, lo cual evidencia que algunos autómatas tienen la propiedad de pseudoaleatoriedad. Esta propiedad fue revisada, medida y verificada para algunas reglas en los trabajos de Fernández Fraga y Rangel Mondragón (2014), Mattos y Moreira (2004) y Villareal Zapata y Ordaz Salazar (2011).
- Clase 4, autómatas celulares complejos: su comportamiento se enmarca dentro de las clases 2 y 3, pero su evolución es más lenta.

Además de la pseudoaleatoriedad, algunos autómatas celulares elementales presentan la característica de reversibilidad, la cual consiste en que ciertas reglas tienen una compañera que invierte el efecto producido por la iteración de la primera, es decir que vuelve a la configuración anterior. Los autómatas de Wolfram solamente tienen 4 pares de reglas reversibles entre sí: 15-85, 51-51, 170-240 y 204-204 (Zhang *et al.*, 2018). En la figura 23, se presenta un ejemplo en el que la primera fila corresponde a la configuración inicial (edén), a esta se aplica la regla 15 en tres ocasiones para obtener tres nuevos estados, al último de ellos se aplica la regla 85 tres veces y se retorna nuevamente a la configuración inicial.

Figura 23. Reglas reversibles 15-85 de Wolfram



Fuente: elaboración propia.

Para el caso de los autómatas bidimensionales, aún no es posible determinar si existen reglas reversibles (Kari, 2005a, 2005b).

Autómatas bidimensionales

Los autómatas bidimensionales se caracterizan porque la evolución de las celdas puede darse en dos dimensiones, es decir que puede tener vecinos no solo a la izquierda y a la derecha, sino también arriba y abajo. En este caso, las reglas de evolución deben tener en cuenta ese tipo de vecindad.

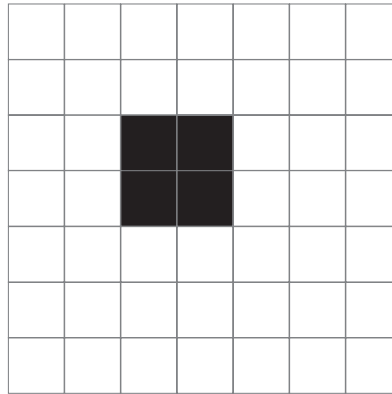
Un caso especial de dichos autómatas es el juego de la vida (GOL por sus siglas en inglés), planteado por John Horton Conway en 1970. Este consiste en lo siguiente: en el espacio bidimensional con bordes periódicos, una célula tiene 2 estados (0: apagado o 1: encendido); si se considera una vecindad de Moore, cada una de las celdas cuenta con 8 vecinos que la rodean, los cuales influyen en su evolución de acuerdo con las premisas que se presentan a continuación (Fernández Fraga y Rangel Mondragón, 2014).

- Cuando la célula esté apagada y tenga 3 vecinos en estado 1, pasará al estado 1 en la siguiente generación.
- Cuando la célula esté encendida y tenga 2 o 3 vecinos encendidos, se mantiene su estado; en caso contrario, cambia a estado 0 en la siguiente generación.

Dependiendo de las condiciones iniciales, este juego de la vida puede dar lugar a diferentes tipos de configuraciones como estables, osciladores, caminadores y formas de vida inmortales (Rayo, 2010).

Estables: el autómata evoluciona hasta obtener siempre los mismos valores para las celdas. En la figura 24, se muestra un ejemplo en el que directamente la ubicación de las celdas hace que, de acuerdo con las reglas del juego de la vida, la configuración se mantenga igual para todas las iteraciones siguientes.

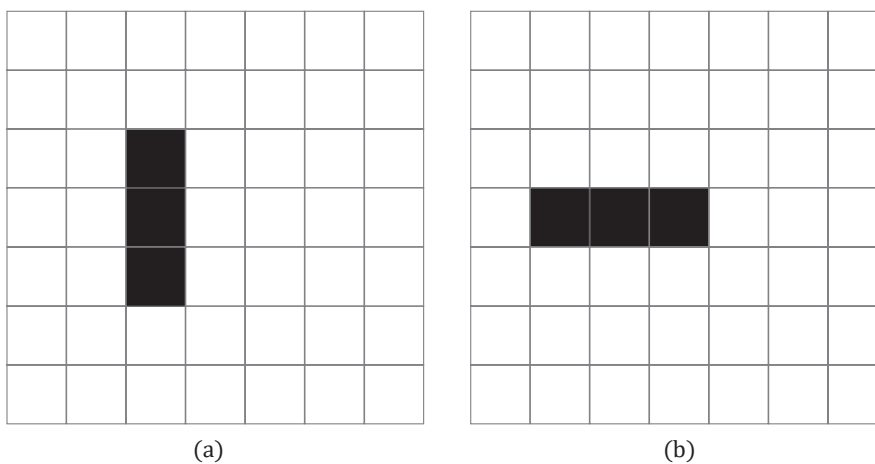
Figura 24. Ejemplo de configuración estable



Fuente: elaboración propia.

Osciladores: el conjunto de celdas vecinas cambia de configuración de manera cíclica y retorna cada cierto número de iteraciones al estado inicial. Algunos osciladores pasan por varias configuraciones antes de regresar a la inicial. En la figura 25, se presenta el caso más sencillo de oscilador, el cual consiste en que la figura 25(a) pasa a la figura 25(b) en la siguiente iteración, esta última vuelve al estado de la figura 25(a) en la siguiente iteración y así sucesivamente.

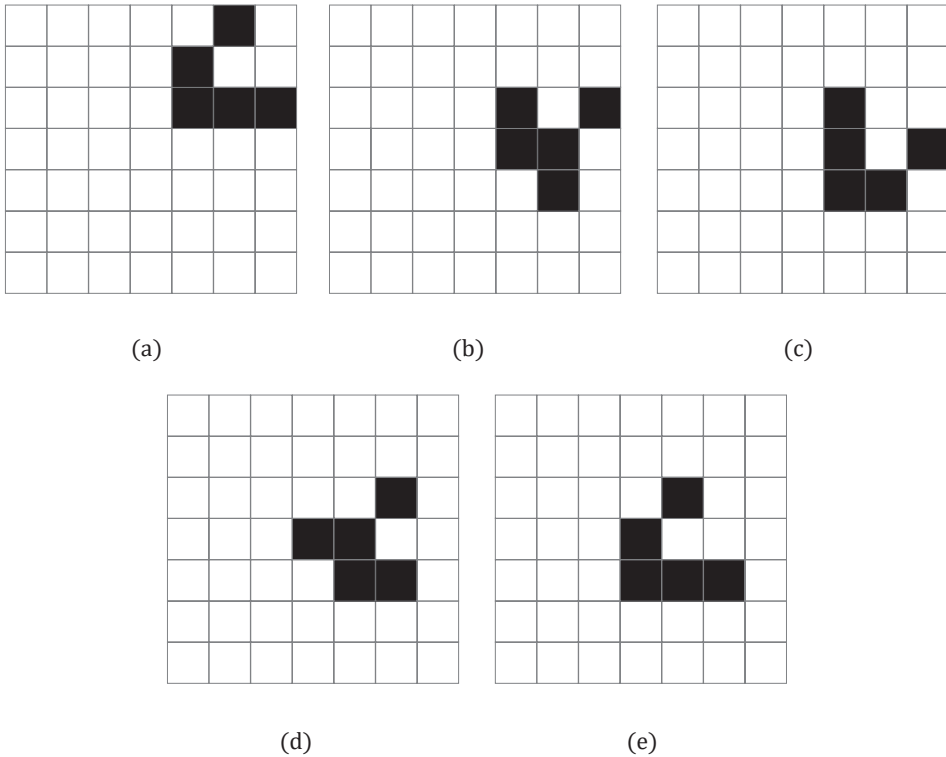
Figura 25. Ejemplo de configuración de un oscilador



Fuente: elaboración propia.

Caminadores: la configuración de las celdas cambia de manera que existe un desplazamiento de estas sobre la región en la cual se encuentran. La iteración repetida de este desplazamiento genera la ilusión de movimiento. En este caso, también se podría pensar en considerarlos osciladores porque retornan a su configuración inicial; sin embargo, la diferencia es que esta configuración estará desplazada sobre la región en la que se llevan a cabo las iteraciones. En la figura 26, se presenta un ejemplo sencillo de caminador en el que la configuración inicial de la figura 26(a), después de las cuatro iteraciones de las figuras 26(b), 26(c), 26(d) y 26(e), aparecerá de nuevo pero desplazada como se muestra en la figura 26(e).

Figura 26. Ejemplo de configuración de un caminador

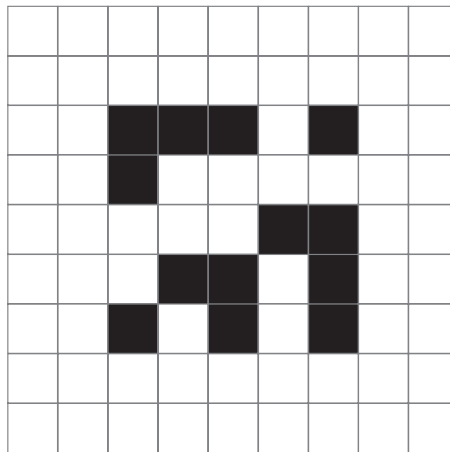


Fuente: elaboración propia.

Formas de vida inmortales: la configuración evoluciona creando indefinidamente otras formas que involucran los tipos mencionados anteriormente.

Algunas configuraciones dan la sensación de generar *células vivas* que se desplazan por la región. La figura 27 irá cambiando y creciendo indefinidamente, en ningún momento, volverá a su configuración inicial y cada vez ocupará un mayor espacio en la región.

Figura 27. Ejemplo de configuración de una forma de vida inmortal



Fuente: elaboración propia.

Los autómatas celulares tienen aplicaciones en diversas áreas, como se evidencia en Łach y Svyetlichnyy (2014), Navarro *et al.* (2020) y Zaheri *et al.* (2020). En la sección Aplicación de autómatas celulares del capítulo 6, se describe un modelo de seguridad orientado a la protección de información sensible, proveniente de videos de cámaras de vigilancia.

Referencias

- Aprendemos matemáticas. (s. f.). *Fractales y autómatas celulares*. <http://www3.gobiernodecanarias.org/medusa/ecoblog/mrodperv/fractales/fractales-y-automatas-celulares/>
- Cano Rojas, A. y Rojas Matas, Á. (2016). Autómatas celulares y aplicaciones. *UNIÓN Revista Iberoamericana de Educación Matemática*, 12(46), 33-48. <https://revistaunion.org/index.php/UNION/article/view/559/308>

- Carvalho, T. I., Carneiro, M. G. y Oliveira, G. M. B. (2019). Improving Cellular Automata Scheduling through Dynamics Control. *International Journal of Parallel, Emergent and Distributed Systems*, 34(1), 115-141. <https://doi.org/10.1080/17445760.2017.1422185>
- Cattaneo, G., Formenti, E., Margara, L. y Mauri, G. (1999). On the Dynamical Behavior of Chaotic Cellular Automata. *Theoretical Computer Science*, 217(1), 31-51. [https://doi.org/10.1016/S0304-3975\(98\)00149-2](https://doi.org/10.1016/S0304-3975(98)00149-2)
- Fernández Fraga, S. M. y Rangel Mondragón, J. (2014, del 10 al 12 de abril). Autómatas celulares y su aplicación en computación. *VIII Congreso Internacional de Ingeniería Mecánica y Mecatrónica*, Puerto Vallarta, México.
- Kari, J. (2005a). Reversible Cellular Automata. In C. De Felice, & A. Restivo (Eds.), *Developments in Language Theory. DLT 2005. Lecture Notes in Computer Science*, 3572, (pp. 57-68). Springer, Berlin, Heidelberg. https://doi.org/10.1007/11505877_5
- Kari, J. (2005b). Theory of Cellular Automata: A Survey. *Theoretical Computer Science*, 334(1-3), 3-33. <https://doi.org/10.1016/j.tcs.2004.11.021>
- Kumar, A. y Kumar Sharma, S. (2022). Information Cryptography Using Cellular Automata and Digital Image Processing. *Journal of Discrete Mathematical Sciences and Cryptography*, 25(4), 1105-1111. <https://doi.org/10.1080/09720529.2022.2072437>
- Łach, Ł. y Svyetlichnyy, D. S. (2014). Frontal Cellular Automata Simulations of Microstructure Evolution during Shape Rolling. *Materials Research Innovations*, 18(sup6), 295-302. <https://doi.org/10.1179/1432891714Z.0000000000971>
- Mattos, T. G. y Moreira, J. G. (2004). Universality Classes of Chaotic Cellular Automata. *Brazilian Journal of Physics*, 34(2a), 448-451. <https://doi.org/10.1590/S0103-97332004000300024>
- Mohammed, J. y Nayak, D. R. (2014). An Efficient Edge Detection Technique by Two Dimensional Rectangular Cellular Automata. *International Conference on Information Communication and Embedded Systems (ICICES2014)*, Chennai, India. <https://doi.org/10.1109/ICICES.2014.7033847>
- Navarro, D., Navarro, M. y Vallejo, I. (2020). Future Urban Seismic Risk Scenarios Using a Cellular Automata Model. *European Planning Studies*, 28(11), 2101-2115. <https://doi.org/10.1080/09654313.2019.1706724>

- Rayo, A. (2010). El juego de la vida: un clásico de John Conway popularizado por Martin Gardner. *Investigación y Ciencia*, (411), 90-91.
- Reyes Gómez, D. A. (2011). *Descripción y aplicaciones de los autómatas celulares*. http://delta.cs.cinvestav.mx/~mcintosh/cellularautomata/Summer_Research_files/Arti_Ver_Inv_2011_DARG.pdf
- Romero, N. (2012). *Dinámica topológica y autómatas celulares: conceptos y propiedades fundamentales*. Editorial Académica Española.
- Sancho Caparrini, F. (2016, 30 de octubre). *Autómatas celulares*. <http://www.cs.us.es/~fsancho/?e=66>
- Vargas-Forero, V. M., Muñoz-Ceballos, L. E., García-Baños, Á. y Jaramillo-Molina, C. (2019). Modelo con autómatas celulares para analizar la accesibilidad peatonal al interior del campus universitario Meléndez de la Universidad del Valle. *Scientia et Technica*, 24(1), 67-75. <https://doi.org/10.22517/23447214.20071>
- Villareal Zapata, E. y Ordaz Salazar, F. C. (2011). Autómatas celulares elementales aplicados a la encriptación de datos. *arXiv*, (1109.5282v1), 1-9. <https://doi.org/10.48550/arXiv.1109.5282>
- Zaheri, M. M., Ghanbari, R. y Afshar, M. H. (2020). A Two-Phase Simulation-Optimization Cellular Automata Method for Sewer Network Design Optimization. *Engineering Optimization*, 52(4), 620-636. <https://doi.org/10.1080/0305215X.2019.1598983>
- Zhang, X., Seo, S.-H. y Wang, C. (2018). A Lightweight Encryption Method for Privacy Protection in Surveillance Videos. *IEEE Access*, 6, 18074-18087. <https://doi.org/10.1109/ACCESS.2018.2820724>

Capítulo 4. Principios de esteganografía

Reseña histórica

La esteganografía tiene sus orígenes en la cultura griega. De acuerdo con la revisión bibliográfica realizada por Amirtharajan y Rayappan (2013), en principio, se utilizaba en planes bélicos. Esto permite afirmar que la esteganografía ha estado presente a lo largo de la historia, ya que intercambiar información secreta sin ser descubierto ha sido un interés común en diferentes ámbitos políticos, religiosos, económicos y sociales.

En la actualidad, su uso es más evidente debido al avance de los computadores y, por ende, de las comunicaciones, lo que ha generado una gama de aplicaciones en diferentes campos como transacciones bancarias, espionaje industrial, espionaje de información privada, implementación de *software* espía y diseño de marcas digitales para proteger derechos de autor. Dichas aplicaciones exigen una mayor cantidad de requerimientos, ya que la información que circula en las redes públicas es cada vez mayor; tal es el caso de imágenes en escala de grises o a color, videos o audios. Esto ha dado origen a lo que se conoce como estegosistemas y ha suscitado que, en las últimas décadas, se haya consolidado una comunidad científica dedicada a plantear técnicas de esteganografía y métodos de detección que satisfagan requerimientos de seguridad, algunas de las cuales se han combinado con técnicas de encriptación para aumentar la protección, utilizando como elementos portadores imágenes, audios o videos.

Las técnicas de esteganografía se interesan por el intercambio de información a través de canales de comunicación encubiertos mediante el cumplimiento de tres estándares básicos: cantidad de información que se puede

ocultar, probabilidad de detección por parte de un estegoanalista y robustez. Este último tiene que ver con la cantidad de alteraciones dañinas que el medio puede soportar sin que haya pérdida de información.

Antiguamente, las técnicas de esteganografía eran fundamentalmente manuales, como uso de tintas, luz ultravioleta, partituras de música, dibujos geométricos, imágenes anamórficas y reducción fotográfica; también se ocultaba información directamente en alguna parte del cuerpo humano. En la actualidad, estas técnicas se apoyan en la computación y en principios matemáticos como autómatas celulares o sistemas dinámicos que exhiben caos.

Cabe resaltar que la esteganografía se ha aplicado con fines negativos para la sociedad. Un ejemplo es el ataque que se realizó a Estados Unidos el 11 de septiembre del 2001. Este acto terrorista fue posible gracias a la esteganografía, ya que los organizadores del plan transmitían imágenes con información acerca del delito en imágenes de pornografía infantil y, así, daban pistas de cómo se ejecutarían las instrucciones para consolidar tal acto (Villa Estrada y Jaramillo Pérez, 2015).

Muchos de los acosos informáticos son una realidad porque, para los Gobiernos, no es computacionalmente viable analizar todas las posibles aplicaciones de esta técnica, por lo que los grandes desarrolladores aprovechan esta falencia. El *malware* es un procedimiento esteganográfico utilizado para ocultar programas sofisticados dentro de, por ejemplo, publicidad, con el fin de robar información importante o satisfacer objetivos delictivos como robo de cuentas bancarias o asaltos industriales, lo que puede ocasionar que se detenga toda la producción de una compañía (Villa Estrada y Jaramillo Pérez, 2015).

El mecanismo aplicado es hacer creer al usuario que se trata de algo inofensivo anunciándole, por ejemplo, que se ha convertido en un afortunado ganador de un premio llamativo. Esto le genera seguridad y lo motiva a descargar un *software* que tiene como fin encontrar puntos débiles del sistema para alojarse allí y extraer información importante para realizar fraude, hacer espionaje o, incluso, dejar fuera de servicio un equipo.

Debido a esto y gracias al avance tecnológico, las aplicaciones esteganográficas cobran mayor interés dada la gran cantidad de información privada

que circula por internet y a la necesidad de establecer mecanismos para su protección. En ese sentido, la esteganografía explota las limitaciones de la percepción humana, como las del ojo y las del oído.

Fundamentación teórica

Algunas referencias útiles que se pueden consultar para ampliar la temática desarrollada en esta sección son Arias del Cerro (2021), Kipper (2007), Lakshmi Sirisha y Chandra Mohan (2021), Shih (2017) y Velásquez Moreira *et al.* (2017).

La palabra esteganografía (*steganography* en inglés) se deriva de las raíces griegas *steganos*, que significa cubierta, y *graphein*, que significa escribir. La esteganografía digital es el arte y la ciencia de ocultar información en canales encubiertos para evitar la detección de los mensajes. Esta disciplina ha tenido un desarrollo importante en los últimos treinta años (Shih, 2017).

El propósito de la esteganografía es evitar sospechas sobre la transmisión de un mensaje secreto en un medio portador, el cual junto con el mensaje oculto se denomina objeto estego. Se trata de que, a toda costa, el portador y el objeto estego sean perceptiblemente indistinguibles. Intentar descubrir si existe un mensaje secreto oculto en un medio genera lo que se conoce en la literatura como estudio estegoanalítico o estegoanálisis.

Las técnicas de esteganografía modernas son cada vez más poderosas en términos de seguridad y robustez. Existen muchas herramientas de *software* que permiten ocultar mensajes en imágenes digitales, videos o audios.

Tipos de esteganografía

Según Shih (2017), en la actualidad, la esteganografía se clasifica en técnica, lingüística y digital. La *esteganografía técnica* oculta la información mediante el uso de tinta invisible o compuestos orgánicos como leche, vinagre, orina y jugo, entre otros. Las tintas invisibles modernas son fluorescentes bajo la luz ultravioleta y pueden ser una estrategia para detectar falsificaciones. Cabe resaltar que se puede encubrir cierta información en un único punto utilizando dibujos geométricos, es decir, usando puntos, líneas o triángulos de un dibujo. Actualmente, la aplicación de estas ideas se puede ver en impresoras que, después de haber generado un documento, permiten que este

sea identificado debido a que ubican puntos invisibles no detectables por el ojo humano.

De otra parte, la *esteganografía lingüística*, también llamada esteganografía textual, tiene como base el lenguaje natural escrito para ocultar información y se puede categorizar en dos ramas: códigos abiertos y semagramas. Los semagramas ocultan mensajes mediante símbolos o signos visuales. En relación con esta estrategia y aprovechando el avance de la tecnología, se pueden utilizar computadores para facilitar que el mensaje secreto sea menos notorio, por ejemplo, se puede alterar el tamaño de ciertas letras específicas, hacer que sus ubicaciones sean un poco más altas o más bajas para enmascarar el mensaje, adicionar espacios en ubicaciones específicas del texto o ajustar el espaciado entre líneas. En general, los semagramas no son técnicas seguras de esteganografía.

Por su parte, los códigos abiertos utilizan letras, palabras o frases de texto o comunicación para esconder mensajes, por ejemplo, ciertas letras situadas en determinadas posiciones de un documento escrito o de una canción pueden constituir un mensaje secreto, el cual no es obvio para un lector promedio. La esteganografía lingüística fue ampliamente utilizada durante la Segunda Guerra Mundial (Shih, 2017). En la actualidad, esta técnica fusiona principios de la esteganografía con la lingüística computacional para diseñar procedimientos de ocultamiento públicos no triviales.

La *esteganografía digital* tiene como objetivo ocultar mensajes secretos dentro de medios digitales como imágenes, audios o videos, lo cual ha tenido un auge en las últimas décadas con aplicaciones relacionadas con espionaje industrial y distribución de material protegido por derechos de autor. La esteganografía digital es prometedora, ya que el tamaño de los medios portadores es muy grande.

La tecnología informática moderna junto con la conectividad masiva permite a las personas, las instituciones o el sector empresarial diseñar páginas web que contienen información secreta destinada a terceros. Aunque cualquier persona puede acceder a la página, para acceder a la información secreta, se requiere de un *software* específico con las claves respectivas para realizar el proceso. Si se desea aumentar una capa de seguridad, se puede

fusionar la esteganografía con la criptografía; la situación es análoga a cuando se requiere guardar algún documento o mercancía valiosa en una caja fuerte: para mejorar la seguridad del proceso, se oculta dicha caja en un lugar difícil de visualizar y acceder (Shih, 2017).

Todos los archivos digitales pueden ser usados para hacer esteganografía, aunque los más apropiados son aquellos que tienen un alto grado de redundancia como imágenes, videos y audios. En este contexto, la redundancia se entiende como el número de bits necesarios en un archivo que se tomará como portador para incrustar los bits de un mensaje oculto sin ser detectado por el ojo o el oído humano (Shih, 2017).

Dentro de la esteganografía digital, se encuentran las siguientes tendencias:

- Ocultar mensajes en sistemas de archivos y en sistemas operativos: estas estrategias se utilizaron hace unos años para ocultar troyanos, esconder medidas de protección de un *software* u ocultar la partición de datos.
- Esconder mensajes secretos en una partición oculta de un disco duro: desde luego que esta ubicación es invisible, aunque se puede acceder mediante la configuración del disco y otras herramientas.
- Ocultar mensajes valiéndose de la estructura lógico-física de los soportes de almacenamiento: esto es posible porque la estructura final de los datos persistentes en el dispositivo puede facilitar el diseño de estrategias para ocultar información en dicha unidad.
- Ocultar mensajes utilizando las tecnologías web y lenguajes de marcado: debido a la gran cantidad de información intercambiada en la web por estas tecnologías, se buscan procedimientos para ocultar la información que dificulten su detección por un estegoanalista.
- Encubrir mediante protocolos de comunicación como TCP/IP: en la mayoría de estos, existen debilidades o consideraciones que no están estrictamente definidas en el diseño, las cuales posibilitan crear canales ocultos de información que pasan desapercibidos. Esto se puede realizar manipulando el formato base de los paquetes a enviar mediante el uso de campos reservados o campos que no se validan.

- Encubrir información mediante protocolos de red, aprovechando la secuencia de números en los segmentos de control de transmisión y de identificación de los paquetes.

Existen diversas técnicas para aplicar esteganografía digital, como (i) el *uso de múltiples claves* generadas para diferentes tipos de codificación en cada fragmento (de tamaño arbitrario) del mensaje; (ii) la *esteganografía en capas*, que busca establecer una relación lineal entre los elementos del mensaje oculto, es decir, se crea una relación de dependencia entre un elemento y su compañero anterior, y (iii) la *adición de ruido*, en este caso, se añade ruido dentro del proceso de esteganografía, es decir, se cambian bits aleatorios adicionales a los modificados para insertar el mensaje. Sin embargo, una de las técnicas más conocidas y utilizadas es la de sustitución de bits menos significativos (LSB), además de la dispersión de mensajes, el enmascaramiento y filtrado, y las funciones de procesamiento de imágenes.

Técnica de sustitución del bit menos significativo. Estego-algoritmo

La técnica de sustitución del bit menos significativo (LSB por sus siglas en inglés) es el mecanismo esteganográfico más empleado, dada su facilidad de implementación, cuando se utilizan medios portadores como imagen, audio o video. Esto se debe a la gran cantidad de espacio disponible para ocultar información.

Existe un resultado bastante alentador con respecto a la cantidad de información que se puede ocultar en un archivo ejecutable. Los autores del artículo “A novel steganographic methodology for high capacity data hiding in executable files” presentan una metodología novedosa que ilustra cómo se puede ocultar una gran cantidad de datos secretos dentro de un archivo ejecutable. Esto es posible gracias al alto grado de redundancia en este tipo de archivos (Kumar Tiwari y Sahoo, 2011).

Una imagen digital es un conjunto de unidades denominadas píxeles, con valores que indican los diferentes colores visibles. Dependiendo de la resolución y de la codificación que se emplee para el formato gráfico en concreto, un pixel se representa por uno o más octetos en dicho formato. En una imagen en escala RGB, 24 bits de resolución hacen referencia a que cada

pixel está representado por tres grupos de 8 bits, es decir, cada grupo puede tomar valores entre 0 y 255 que definen la intensidad del color rojo, verde o azul, cuya combinación puede generar más de dieciséis millones de colores (Kaur *et al.*, 2014).

Cuando se aplica sobre una imagen la técnica LSB, se convierte en una estrategia de sustitución que reemplaza el bit menos significativo de cada pixel por el bit del mensaje que se desea ocultar (Amirtharajan y Rayappan, 2013). La elección de los pixeles a utilizar dependerá de la variante de la técnica LSB a utilizar, dicha elección se puede hacer de acuerdo con una de las siguientes estrategias:

- Sustitución secuencial
- Sustitución pseudoaleatoria
- Criterio matemático de selección

Independientemente de la estrategia que se emplee para seleccionar los pixeles a modificar, el principio que rige el método LSB es que los bits menos significativos de los pixeles de una imagen determinan pequeños cambios en el color. Por consiguiente, su modificación conlleva a introducir cambios mínimos en la imagen que no son percibidos por el ojo humano, de modo que dichas modificaciones no aportan información adicional a un analista.

En ocasiones, para ocultar la información en una imagen, es posible utilizar más de un bit por pixel; sin embargo, se debe tener en cuenta que los cambios en los bits más significativos pueden generar transformaciones que sean evidentes para el ojo humano, especialmente, si se afecta significativamente la capa azul (Shih, 2017). En caso de que se requiera mayor capacidad para ocultar un mensaje, se puede recurrir a una imagen portadora de tamaño superior.

Técnicas esteganográficas para controlar piratería informática y derechos de autor

Para ayudar a controlar la piratería informática y salvaguardar los derechos de autor, se utilizan básicamente dos técnicas: marcas de agua sobre papel y marcas de agua digitales. Las marcas de agua sobre papel se utilizan, por

ejemplo, para garantizar la autenticidad, la autoría, la procedencia y la fecha de publicación de un libro o documento impreso. Asimismo, en el ámbito económico, se emplean marcas de agua sobre los billetes para reducir la posibilidad de que sean falsificados (Shih, 2017).

Las marcas de agua digitales han tomado mucha importancia debido a la necesidad del reconocimiento de autores de libros digitales que circulan en la red. Esta técnica de esteganografía surgió básicamente para proteger derechos de autor en formatos como documentos, videos, imágenes o audios. El objetivo es crear una marca que sea inseparable de los archivos y que indique a quién pertenece la autoría, la propiedad y la distribución, con el fin de evitar plagio. Esta herramienta consiste en insertar un código directamente al archivo, bien sea imagen, audio o video, que funciona como un identificador asociado al autor (Orúe López, 2002).

Clasificación de la esteganografía digital

Existen tres tipos de esteganografía digital: pura, de clave privada y de clave pública. En analogía con la criptografía, la esteganografía de clave privada y la de clave pública también se denominan simétrica y asimétrica, respectivamente.

En la esteganografía pura, la robustez del sistema recae en los algoritmos implementados para ocultar y extraer la información, que solo el emisor y el receptor del mensaje deberían conocer. Este tipo de esteganografía es seguro siempre y cuando el medio por el cual se transmite el mensaje no sea vulnerable a atacantes expertos, con habilidades en esteganografía. Sin embargo, en escenarios más prácticos, este método se combina con criptografía, ya que por sí solo podría resultar débil y desastroso para el propósito que se tiene (Villa Estrada y Jaramillo Pérez, 2015).

La esteganografía de clave privada es la fusión de esteganografía pura y criptosistemas de clave simétrica para cifrar el mensaje antes de iniciar el proceso de esteganografía. La esteganografía de clave pública utiliza dos claves, por lo que no se requiere un intercambio previo de estegoclave. Una de ellas se utiliza en la inserción del mensaje secreto y la otra es la clave pública empleada para reconstruir el mensaje oculto (De la Cruz Franco, 2017).

Herramientas de esteganografía disponibles en la web

El *software* para esteganografía digital moderna emplea algoritmos sofisticados que buscan ocultar información de forma imperceptible. Esto representa un reto importante en la actualidad, debido a la gran cantidad de herramientas de esteganografía digital disponibles gratuitamente en internet que se pueden utilizar para ocultar cualquier archivo digital dentro de otro, lo cual facilita su uso con fines negativos.

Una lista de más de cien herramientas fue recopilada por el investigador Neil F. Johnson (2012) en su página web <http://www.jjtc.com/Steganography/tools.html>, la cual es actualizada constantemente gracias al trabajo colaborativo de distintos usuarios.

Villa Estrada y Jaramillo Pérez (2015) describen algunas de las herramientas disponibles en la web, como el *software DeepSound*. Este es una herramienta gratuita para Windows, diseñada con el fin de ocultar información dentro de archivos MP3 con un cifrado *AES de 256 bits* (Velasco, 2015).

Descripción de técnicas de esteganografía digitales aplicadas en archivos de audio

Velásquez Moreira *et al.* (2017) realizan una descripción de las principales técnicas de esteganografía aplicadas a archivos de audio e imagen; identifican su funcionamiento, fortalezas y debilidades; resaltan el hecho de que la técnica LSB es la más débil, y especifican otras técnicas más seguras, dependiendo del medio portador que se esté utilizando. Se destaca que la esteganografía es mucho más sofisticada en medios digitales (audio, imagen y video) que en otros dominios como los canales de red (Ker *et al.*, 2013). Esto ha motivado un alto grado de investigación con este enfoque y ha generado una gran cantidad de estrategias aplicables a dichos archivos, lo que ha dado paso a nuevos resultados.

En el contexto de la esteganografía de audio, dependiendo de las herramientas matemáticas que se utilicen, las estrategias de clasifican en:

- Dominio temporal
- Dominio de frecuencia

- Dominio Wavelet
- Dominio de códecs

El *dominio temporal*, también denominado espacial, está compuesto por las siguientes técnicas: *Low Bit Encoding*, también conocida como *Least Significant Bit* (LSB), y *Echo Hiding* (Malhotra y Tahirramani, 2014).

La técnica LSB consiste en utilizar una clave generada por un codificador de marcas de agua para elegir muestras de audio extraídas del audio original, las cuales se tomarán como punto de partida para seleccionar los bits en los que se ocultará la información. Por otra parte, la técnica *Echo Hiding* incrusta información dentro de una señal de audio y basa su funcionamiento en encontrar *huecos* dentro de los rangos de percepción del sistema auditivo humano, donde se ocultarán los datos a fin de que estos tengan una degradación mínima con respecto a los datos originales; esto permite que el cambio en el audio sea más difícil de percibir por el oído humano (Velásquez Moreira *et al.*, 2017).

En el *dominio de frecuencia*, según la clasificación realizada por Antony *et al.* (2012), se tienen las técnicas: *tone insertion*, *phase coding*, *amplitude modification* y *spread spectrum*.

En la técnica *tone insertion*, el audio se divide en segmentos de 16 ms de duración, se calcula la potencia de cada segmento y se incrusta únicamente un bit de los datos en el audio original. Esta técnica se encarga de conseguir la imperceptibilidad auditiva de los tonos de baja potencia (Velásquez Moreira *et al.*, 2017). Aunque es resistente a ataques de filtro *low-bass* y truncado de bits, una de sus limitantes es la baja capacidad de incrustación de datos, lo cual permite que los tonos insertados sean fáciles de detectar (Djebbar *et al.*, 2012).

La técnica *phase coding* incrusta información en la fase de la señal portadora y tiene un buen rendimiento en relación con la fidelidad del dato oculto. Sin embargo, una de sus limitantes es la poca capacidad que tiene para ocultar, ya que embebe de 16 a 32 bits de datos en archivos de audio de 1 segundo de duración (Malik *et al.*, 2007).

En la técnica *amplitude modification*, se realiza una traducción del algoritmo dentro del código. Dado que lo más relevante es el ordenamiento de

las principales amplitudes, se utilizan tres secciones de 512 *frames*, en los cuales es posible ocultar separadamente la información dentro de cada canal; sin embargo, existe el riesgo de perder datos en el caso de audios mono. Este método presenta una gran debilidad en términos de audición (Jenkins y Martina, 2009; Velásquez Moreira *et al.*, 2017).

Por último, en el caso de *spread spectrum*, existen dos mecanismos de trabajo: los saltos de frecuencia, donde el código de un pseudoruido se utiliza de manera pseudoaleatoria para ayudar al cambio de transmisión periódica de frecuencia, y la secuencia directa, en la que el código de pseudoruido se usa para modular secuencialmente la señal. Una ventaja de esta técnica es la dificultad que presenta para que la transmisión sea detectada (Velásquez Moreira *et al.*, 2017).

El *dominio Wavelet* tiene propiedades de resolución múltiple, lo cual hace que sea apropiado para el dominio de frecuencia. Además, tiene la ventaja de que, al aplicarse una transformación inversa, se reconstruye la estegano-señal (Antony *et al.*, 2012).

El *dominio de códecs* o dominio de codificador está conformado por las técnicas *codebook modification* y *bistream hiding*. Estas se emplean para incrustar información dentro de archivos de audio enviados en tiempo real, para lo cual utilizan códecs de voz como AMR, ACELP o SILK. Una de las fortalezas de estas técnicas es su robustez; sin embargo, su debilidad radica en que tienen una baja tasa de incrustación (Djebbar *et al.*, 2012).

Descripción de técnicas de esteganografía digitales aplicadas en archivos de imágenes

Las técnicas de esteganografía cuyo elemento portador es una imagen, dependiendo del fundamento matemático que se emplee, se pueden clasificar en (Mehdi Hussain y Mureed Hussain, 2013):

- Dominio espacial
- Dominio de transformación
- Distorsión
- Enmascaramiento y filtrado

En el *dominio espacial*, existen varios métodos entre los que se destacan: la técnica LSB, *RGB based steganography*, *Pixel Value Differencing (PVD)* y *Mapping based steganography*.

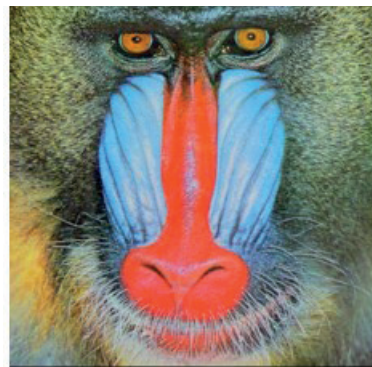
Una vez más, para el caso de una imagen como elemento portador, la técnica LSB es la más simple, cuenta con baja complejidad computacional y tiene una alta capacidad para ocultar información. Esto hace que sea insegura, ya que se pueden realizar ataques para obtenerla (Kaur *et al.*, 2014; Singla y Juneja, 2014).

Como parte de un ejercicio académico (que se abordará con mayor profundidad en el capítulo 6), utilizando la técnica LSB junto con la detección de bordes por medio de autómatas celulares y el uso del atractor caótico de Tinkerbell, se consolidó un modelo esteganográfico con el objetivo de ocultar un texto en una imagen digital RGB. A manera de ejemplo, en la figura 28(b), se presenta el resultado de ocultar un mensaje de 2979 caracteres en la imagen de la figura 28(a). La estegoimagen de la figura 28(b) presenta cambios imperceptibles para el ojo humano, situación deseable en todo sistema esteganográfico.

Figura 28. Resultado modelo esteganográfico con imágenes digitales



(a) Imagen original
(portadora)

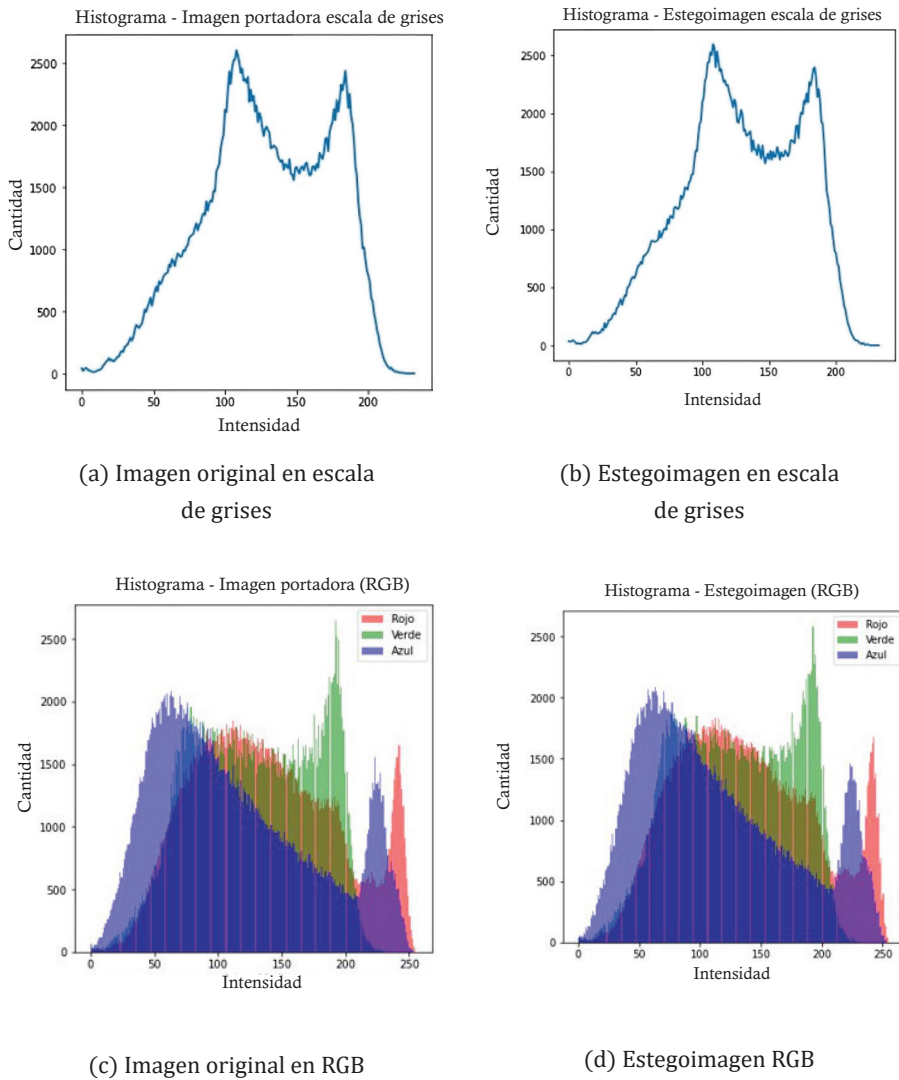


(b) Estegoimagen

Fuente: elaboración propia a partir de Weber (1981).

Otra manera de corroborar la similitud de las imágenes anteriores es a través de los histogramas de frecuencia. Estos se presentan tanto en escala de grises como en RGB en la figura 29, donde se destaca la alta coincidencia entre las imágenes de las figuras 29(a) y 29(b) para escala de grises y entre las imágenes de las figuras 29(c) y 29(d) para RGB.

Figura 29. Histogramas de frecuencia correspondientes a las imágenes de la figura 28



Fuente: elaboración propia.

Por su parte, la técnica *RGB based steganography* utiliza los píxeles de una imagen RGB como cubierta y usa un canal para indicar los datos secretos de los otros dos, el canal de indicación cambia de un píxel a otro de manera aleatoria dependiendo de los píxeles de la imagen. Aunque esta técnica tiene mayor capacidad para ocultar información que la LSB, conserva el mismo nivel de seguridad (Gutub, 2010).

En cuanto a la técnica *Pixel Value Differencing* (PVD), para su aplicación, es necesario partir la imagen portadora en bloques de dos píxeles consecutivos que no se traslapen. Luego, se calcula la diferencia entre los valores de cada par de píxeles; estos valores se distribuyen en rangos que determinan el número de bits pertenecientes a los datos secretos que se incrustarán en el par de píxeles elegido (Wu y Tsai, 2003). Aunque con PVD se pueden ocultar más datos en parejas de píxeles con diferencias más grandes, esta modificación conlleva la pérdida de calidad de la imagen (Shen y Huang, 2015).

Por otra parte, la técnica *Mapping Based Steganography* permite ocultar datos dentro de imágenes en escala de grises. Para esto, se seleccionan los píxeles de la imagen por medio de funciones matemáticas de acuerdo con la intensidad del píxel semilla (Bhattacharyya *et al.*, 2011).

El *dominio de transformación* se subdivide dependiendo del tipo de transformada matemática que se aplique de la siguiente manera: transformada discreta de Fourier (DFT), transformada discreta de cosenos (DCT) y transformada discreta Wavelet (DWT). Estas técnicas de esteganografía son más recomendables, ya que trabajan en este dominio y tienen la posibilidad de ocultar la información en áreas que están menos expuestas a la compresión, recorte y procesamiento de imagen (Hussain y Hussain, 2013).

La técnica DFT está basada en transformaciones matemáticas que convierten los píxeles, de tal manera que da el efecto de difundir la ubicación de los valores de dichos píxeles sobre una parte de la imagen (Morkel *et al.*, 2005). Esta técnica se ha mejorado aplicando *Fractional Fourier Transform* (FrFT), la cual es una generalización de DFT (Soni *et al.*, 2013).

La técnica DCT incrusta el mensaje secreto en el bit menos significativo del coeficiente del coseno discreto aplicado a la imagen portadora. Para esto,

se descompone la imagen en bloques de 8x8 píxeles de derecha a izquierda y en dirección de arriba hacia abajo (Velásquez Moreira *et al.*, 2017).

La técnica DWT se utiliza para descomponer la imagen en pequeñas ondas denominadas ondículas (Velásquez Moreira *et al.*, 2017).

Por otra parte, la técnica de *distorsión* se caracteriza porque, para la decodificación, es necesario tener conocimiento de la imagen original de cobertura, con el fin de determinar si existen diferencias entre la imagen y la estegoimagen (Velásquez Moreira *et al.*, 2017).

La técnica de *enmascaramiento y filtrado* crea máscaras en una imagen que no son detectables por el ojo humano, los datos a incrustar se ocultan en la parte visible de la imagen. Esta técnica es más eficaz que la LSB y es aplicable solo para imágenes de 24 bits o imágenes en escala de grises (Hussain y Hussain, 2013; Velásquez Moreira *et al.*, 2017).

Estegoanálisis

El estegoanálisis es el proceso de detección de esteganografía. Los sistemas estegoanalíticos se pueden caracterizar en el dominio espacial o en el dominio de frecuencia. Los sistemas estegoanalíticos espaciales son adoptados para verificar que no haya pérdidas en las imágenes comprimidas mediante el análisis estadístico. Para imágenes comprimidas con pérdida como JPEG (*Joint Photographic Experts Group*), se hacen análisis estadísticos en el dominio de la frecuencia.

Desde hace varias décadas, se conocen diversos trabajos acerca del estegoanálisis. Entre estos se encuentran aquellos reportados en 1999, que se basaban en ataques visuales a partir de la inspección de los planos de bit más bajos en las estegoimágenes (Westfeld y Pfitzmann, 1999).

De otra parte, existen técnicas para estegoanálisis de imágenes que han sido potencialmente sometidas a procedimientos de esteganografía. Estas parten de la hipótesis de que los esquemas esteganográficos dejan evidencias estadísticas que se pueden explotar para la detección con ayuda de las características alusivas a la calidad de la imagen y con análisis de regresión. Para esto, se propone un indicador llamado IQM (*image quality measure*) (Avcibas

et al., 2003). Los resultados de las simulaciones muestran que su propuesta es capaz de distinguir entre imágenes portadoras e imágenes estego.

Existe una metodología para aplicar ataques a esquemas esteganográficos basados en imágenes JPEG. Se parte de la descompresión de la imagen estego en formato JPEG, se distorsiona geométricamente y, luego, se vuelve a comprimir. Esto rompe la estructura de los coeficientes de la transformada discreta de cosenos (Fridrich *et al.*, 2003a).

Básicamente, existen dos métodos para detectar archivos modificados. Uno de ellos es el *image quality measure* (IQM), el cual se basa en un análisis visual que consiste en comparar un archivo sospechoso con la copia original. Este pretende revelar la presencia de comunicación secreta a través de la inspección, ya sea visualmente o con la ayuda de un sistema informático, mediante la descomposición de la imagen en capas de bits. Aunque este método es simple, no es muy efectivo porque, por lo general, la copia original no siempre está disponible (Shih, 2017).

El otro método se llama análisis estadístico, el cual detecta cambios en los patrones de píxeles y la distribución de frecuencias. Esto permite inferir si una imagen se ha modificado al comprobar si sus propiedades estadísticas se desvían y encontrar incluso ligeras alteraciones en el comportamiento estadístico causado por la incrustación esteganográfica (Shih, 2017).

Es necesario que las técnicas estegoanalíticas se actualicen y evolucionen continuamente para combatir los métodos esteganográficos nuevos y más sofisticados. Sin embargo, hay ciertas reglas que los investigadores adoptan cuando buscan pistas que sugieren el uso de esteganografía. Estas incluyen las capacidades técnicas del propietario de la computadora, pistas de *software*, archivos de programa, archivos multimedia y el tipo de delito (Shih, 2017).

Propiedades estadísticas de la imagen

Existen propiedades estadísticas para la percepción humana de las imágenes naturales, los cambios dinámicos en la longitud de onda de la luz se traducen en estímulos visuales que conllevan a que los colores rojo, naranja, amarillo, verde y azul se perciban como violeta. Las sensibilidades visuales humanas están estrechamente relacionadas con el color, por ejemplo, los seres humanos tienen más sensibilidad al azul que al verde y al rojo (Shih, 2017).

El brillo, el matiz y la saturación son tres elementos indispensables para describir un color. Cuando el ojo humano se acostumbra a un tono específico, tiende a distorsionar la percepción de otros tonos, por ejemplo, cuando el ojo se ha adaptado al color verde y se enfrenta al color gris, ve a este como si fuera púrpura. El cifrado de mensajes ocultos también dificulta la detección porque los datos encriptados generalmente tienen un alto grado de aleatoriedad y existen unos y ceros con la misma probabilidad. De este modo, además de la detección, la recuperación de mensajes ocultos parece ser un problema complicado porque requiere, además del análisis esteganográfico, el conocimiento del algoritmo criptográfico o la clave de cifrado (Farid, 2001; Fridrich *et al.*, 2003).

Existen cuatro métodos estegoanalíticos: regular-singular (RS), pares de valores, análisis de pares y pares de muestras. Estos aprovechan los cambios estadísticos al modelar modificaciones de ciertos valores en función del porcentaje de píxeles con esteganografía incrustada (Shih, 2017). Una descripción detallada de estos métodos se puede consultar en los trabajos de Dumitrescu *et al.* (2003), Fridrich *et al.* (2001, 2003c) y Westfeld y Pfitzmann (2000).

Los análisis estadísticos para imágenes incluyen: media, varianza y el test chi cuadrado χ^2 (Wayner, 2002).

Sistemas estegoanalíticos visuales

Kessler (2004) presenta una visión general de la esteganografía para los examinadores forenses informáticos. Las estrategias de estegoanálisis suelen seguir el método que utiliza el algoritmo de esteganografía. Una forma simple es inspeccionar visualmente el portador y el medio de esteganografía, lo cual concierne a los sistemas estegoanalíticos visuales (VSS por sus siglas en inglés), ya que muchas técnicas incorporan mensajes secretos en el dominio espacial y seleccionan los bits del mensaje en el portador independientemente de su contenido. La mayoría de los sistemas esteganográficos incluyen bits del mensaje a incrustar de manera secuencial o pseudoaleatoria. Si la imagen contiene ciertas áreas conectadas de color uniforme o de color saturado (es decir, 0 o 255), se puede utilizar la inspección visual para hallar artefactos

sospechosos; si no se encuentran dichos artefactos, se realiza una inspección del plano de bits LSB (Wayner, 2002).

Por otro lado, Westfeld y Pfitzmann (1999) presentan un VSS que utiliza una función de asignación para reemplazo de color denominada filtro visual. Su propósito es eliminar todas las partes de la imagen que contienen un mensaje potencial. El proceso de filtrado se basa en la presunta herramienta esteganográfica que puede atacar al esteganograma del medio portador, extraer los bits de mensaje potencial y producir una ilustración visual.

Sistemas estegoanalíticos basados en *image quality measure*

El método *image quality measure* (IQM) puede caracterizarse por la correlación entre la pantalla y el sistema visual humano (HVS). Con el indicador IQM se pretende minimizar las decisiones subjetivas. El análisis de la calidad de la imagen computacional se orienta hacia el desarrollo de un modelo preciso para describir la calidad de la imagen.

Hay dos tipos de IQM: uno con *full* referencia y otro sin referencia. El IQM con *full* referencia tiene la imagen original disponible para ser comparada con la distorsionada; en este, se aplican diferentes tipos de métricas con el fin de medir la cercanía entre estas (Shih, 2017). En el IQM sin referencia, solo se toma la imagen distorsionada para el análisis y se utilizan operadores de procesamiento como detector de bordes para calcular el índice de calidad de la imagen.

Un trabajo que aporta en la dirección de este estudio fue el desarrollado por Shih (2017). En este, el autor plantea una técnica basada en el IQM para el estegoanálisis de imágenes que han sido sometidas a algoritmos de esteganografía y asume que los esquemas esteganográficos dejan evidencia estadística, la cual puede ser explotada para la detección con la ayuda de características de calidad de la imagen y un análisis de regresión multivariada. Para esto, se han identificado métricas de calidad de la imagen basadas en técnicas de análisis de varianza como un conjunto de características para distinguir entre las portadoras y las estegoimágenes. La clasificación entre las imágenes portadoras y las estegoimágenes se realiza utilizando una regresión multivariante en la métrica de calidad seleccionada y se entrena con base en una estimación de la imagen original.

Referencias

- Amirtharajan, R. y Rayappan, J. B. B. (2013). Steganography-Time to Time: A Review. *Research Journal of Information Technology*, 5(2), 53-66.
- Arias del Cerro, A. G. (2021). *Esteganografía en archivos digitales* [Tesis de pregrado, Universidad Complutense de Madrid]. <https://hdl.handle.net/20.500.14352/5358>
- Avcıbaşı, I., Memon, N. y Sankur, B. (2003). Steganalysis Using Image Quality Metrics. *IEEE Trans Image Process*, 12(2), 221-229. <https://doi.org/10.1109/TIP.2002.807363>
- Bhattacharyya, S., Khan, A., Nandi, A., Dasmalakar, A., Roy, S. y Sanyal, G. (2011). Pixel Mapping Method (PMM) Based Bit Plane Complexity Segmentation (BPCS) Steganography. *2011 World Congress on Information and Communication Technologies*, Mumbai, India. <https://doi.org/10.1109/WICT.2011.6141214>
- De la Cruz Franco, A. (2017). *Implementación de un algoritmo computacional para esteganografía basado en técnicas del bit menos significativo* [Tesis de pregrado, Universidad de Quintana Roo]. <http://hdl.handle.net/20.500.12249/1950>
- Djebbar, F., Ayad, B., Meraim, K. A. y Hamam, H. (2012). Comparative Study of Digital Audio Steganography Techniques. *EURASIP Journal on Audio, Speech, and Music Processing*, (25), 1-16. <https://doi.org/10.1186/1687-4722-2012-25>
- Dumitrescu, S., Wu, X. y Wang, Z. (2003). Detection of LSB Steganography Via Sample Pair Analysis. *IEEE Transactions on Signal Processing*, 51(7), 1995-2007. <https://doi.org/10.1109/TSP.2003.812753>
- Farid, H. (2001). Detecting Steganographic Messages in Digital Images. *Computer Science Technical Reports*, (TR2001-412), 1-9. https://digitalcommons.dartmouth.edu/cgi/viewcontent.cgi?article=1197&context=cs_tr
- Fridrich, J., Goljan, M. y Du, R. (2001, octubre). Reliable Detection of LSB Steganography in Color and Grayscale Images. *2001 Workshop on Multimedia and Security: New Challenges*. <https://doi.org/10.1145/1232454.1232466>
- Fridrich, J., Goljan, M. y Hoge, D. (2003a, 20 de junio). New Methodology for Breaking Steganographic Techniques for JPEGs. *SPIE 5020, Security and Watermarking of Multimedia Contents V*. <https://doi.org/10.1117/12.473142>

- Fridrich, J., Goljan, M., Hoge, D. y Soukal, D. (2003b). Quantitative Steganalysis of Digital Images: Estimating the Secret Message Length. *Multimedia Systems*, 9, 288-302. <https://doi.org/10.1007/s00530-003-0100-9>
- Fridrich, J., Goljan, M. y Soukal, D. (2003c, June 20). Higher-Order Statistical Steganalysis of Palette Images. *SPIE 5020, Security and Watermarking of Multimedia Contents V*. <https://doi.org/10.1117/12.473140>
- Gutub, A. A.-A. (2010). Pixel Indicator Technique for RGB Image Steganography. *Journal of Emerging Technologies in Web Intelligence*, 2(1), 56-64. <http://www.jetwi.us/uploadfile/2014/1226/20141226043213435.pdf>
- Hussain, M. y Hussain, M. (2013). A Survey of Image Steganography Techniques. *International Journal of Advanced Science and Technology*, 54, 1-12. https://www.researchgate.net/publication/271863505_A_Survey_of_Image_Steganography_Techniques
- Jenkins, N. y Martina, J. E. (2009). Steganography in Audio. *Anais do IX Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais*, 269-278. https://www.researchgate.net/publication/228522900_Steganography_in_Audio
- Jisna, A., Sobin, C. C. y Sherly, A. P. (2012). Audio Steganography in Wavelet Domain - A survey. *International Journal of Computer Applications*, 52(13), 33-37. <https://research.ijcaonline.org/volume52/number13/pxc3881810.pdf>
- Johnson, N. F. (2012). *Steganography Software*. <https://www.jjtc.com/Steganography/tools.html>
- Kaur, S., Bansal, S. y Bansal, R. K. (2014). Steganography and Classification of Image Steganography Techniques. *2014 International Conference on Computing for Sustainable Global Development (INDIACom)*, New Delhi, India. <https://doi.org/10.1109/IndiaCom.2014.6828087>
- Ker, A., Bas, P., Böhme, R., Cograñne, R., Craver, S., Filler, T., Fridrich, J. y Pevny, T. (2013, June). Moving Steganography and Steganalysis from the Laboratory into the Real World. *CM IH-MMSEC 2013*, Montpellier, France. <https://hal.science/hal-00836407>
- Kessler, G. C. (2004). An Overview of Steganography for the Computer Forensics Examiner. *Forensic Science Communications*, 6(3). <https://commons.erau.edu/db-security-studies/19>

- Kipper, G. (2007). What is Steganography? In J. Wiles, & R. Rogers, *Techno Security's Guide to Managing Risks for IT Managers, Auditors, and Investigators* (pp. 311-335). Syngress. <https://www.sciencedirect.com/science/article/abs/pii/B9781597491389500168?via%3Dihub>
- Kumar Tiwari, R. y Sahoo, G. (2011). A Novel Steganographic Methodology for High Capacity Data Hiding in Executable Files. *International Journal of Internet Technology and Secured Transactions*, 3(2), 210-222. <https://doi.org/10.1504/IJTST.2011.039779>
- Lakshmi Sirisha, B. y Chandra Mohan, B. (2021). Review on Spatial Domain Image Steganography Techniques. *Journal of Discrete Mathematical Sciences and Cryptography*, 24(6), 1873-1883. <https://doi.org/10.1080/09720529.2021.1962025>
- Malhotra, N. y Tahilramani, N. (2014). Survey on Speech and Audio Steganography Techniques in Temporal, Transform and Coded Domains. *International Journal of Advanced Research in Computer Science and Software Engineering*, 800-809.
- Malik, H. M. A., Ansari, R. y Khokhar, A. A. (2007). Robust Data Hiding in Audio Using Allpass Filters. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(4), 1296-1304. <https://doi.org/10.1109/TASL.2007.894509>
- Morkel, T., Eloff, J. H., Olivier, M. S. (2005, 29 de junio-1 de julio). An Overview of Image Steganography [conference]. *Information Security for South Africa (ISSA)*, Pretoria, South Africa. https://digifors.cs.up.ac.za/issa/2005/Proceedings/Full/098_Article.pdf
- Orúe López, A. B. (2002). Marcas de agua en el mundo real. *Boletín del Criptonomicón Año IV*, (88). <http://hdl.handle.net/10261/8864>
- Shen, S.-Y. y Huang, L.-H. (2015). A Data Hiding Scheme Using Pixel Value Differencing and Improving Exploiting Modification Directions. *Computers and Security*, 8(C), 131-141. <https://doi.org/10.1016/j.cose.2014.07.008>
- Shih, F. Y. (2017). *Digital Watermarking and Steganography: Fundamentals and Techniques*(2.ª ed.). CRC Press. <https://doi.org/10.1201/9781315121109>

- Singla, D. y Juneja, M. (2014). An Analysis of Edge Based Image Steganography Techniques in Spatial Domain. *2014 Recent Advances in Engineering and Computational Sciences (RAECS)*, Chandigarh, India. <https://doi.org/10.1109/RAECS.2014.6799604>
- Soni, A., Jain, J. y Roshan, R. (2013). Image Steganography Using Discrete Fractional Fourier Transform. *2013 International Conference on Intelligent Systems and Signal Processing (ISSP)*, Vallabh Vidyanagar, India. <https://doi.org/10.1109/ISSP.2013.6526882>
- Velasco, R. (2015, 31 de octubre). *DeepSound: cómo ocultar información en archivos de música al estilo Mr. Robot*. RZ redes zone. <https://www.redeszone.net/2015/10/31/deepsound-como-ocultar-informacion-en-archivos-de-musica-al-estilo-mr-robot/>
- Velásquez Moreira, G. M., Molina Sabando, L. A. y Briones Véliz, I. B. (2017). Análisis de técnicas de esteganografía aplicadas en archivos de audio e imagen. *Polo del Conocimiento*, 2(1), 51-67. <https://doi.org/10.23857/pc.v2i1.10>
- Villa Estrada, H. F. y Jaramillo Pérez, J. C. (2015). *Aplicaciones de la esteganografía en la seguridad informática* [Tesis de pregrado, Universidad Tecnológica de Pereira]. <https://repositorio.utp.edu.co/server/api/core/bitstreams/91c94967-c7cc-4b06-a890-b4224e1c96bd/content>
- Wayner, P. (2002). *Disappearing Cryptography: Information Hiding: Steganography and Watermarking* (2.ª ed.). Morgan Kaufmann Publishers Inc.
- Weber, A. (1981). *The USC-SIPI Image Database*. Signal and Image Processing Institute. University of Southern California. <https://sipi.usc.edu/database/>
- Westfeld, A. y Pfitzmann, A. (2000). Attacks on Steganographic Systems. In A. Pfitzmann (Ed.), *Information Hiding. IH 1999. Lecture Notes in Computer Science*, 1768 (pp. 61-76). Springer. https://doi.org/10.1007/10719724_5
- Wu, D.-C. y Tsai, W.-H. (2003). A Steganographic Method for Images by Pixel-Value Differencing. *Pattern Recognition Letters*, 24(9-10), 1613-1626. [https://doi.org/10.1016/S0167-8655\(02\)00402-6](https://doi.org/10.1016/S0167-8655(02)00402-6)

Capítulo 5. Criptografía

Reseña histórica

La criptografía se originó en la Antigüedad debido a la necesidad de enviar información privada de manera oculta y para evitar que observadores no autorizados tuvieran acceso a esta. Este problema de transmisión segura de la información se ha ido adaptando a los diferentes canales de comunicación a lo largo de la historia, particularmente, en épocas de conflictos o guerras. De esta manera, se ha pasado del papel al telégrafo, a la radio, al teléfono de línea fija y a la televisión, hasta llegar a las actuales redes de comunicación (Bingöl, 2022; Lucena López, 2016; Ortega Triguero *et al.*, 2006).

En los últimos años, debido a la gran cantidad de información que se intercambia a través de canales públicos y a la necesidad de evitar que alguien no autorizado pueda manipular y sacar provecho de los datos, lo cual puede afectar la estabilidad emocional, familiar y social de las personas a quienes pertenece dicha información, varios países han regulado su manejo. Sin embargo, estas medidas no han sido suficientes, más aún si se tienen en cuenta los resultados presentados en el *Ninth Annual Cost of Cybercrime Study*, publicado por Accenture (Ponemon Institute LLC y Accenture Security, 2019). Estos evidencian que, durante el 2018, el promedio de pérdidas por cibercrimen ascendió a trece millones de dólares para las empresas de dieciséis sectores de la industria en once países. Asimismo, se encontró que los ciberataques se han incrementado en un 72 % desde el 2013.

Esto ha ocasionado el surgimiento de nuevas técnicas de criptografía basadas en diferentes teorías matemáticas, que han llamado la atención de la comunidad científica con miras a proponer esquemas cada vez más rápidos,

seguros y aptos para aplicaciones prácticas que requieran el ocultamiento eficiente de grandes cantidades de información. Como consecuencia de la pandemia ocasionada por la COVID-19, el uso de herramientas tecnológicas para el intercambio de información ha tenido un crecimiento exponencial. Esto exige el diseño de esquemas de seguridad que eviten el acceso a intrusos con fines ilícitos y con consecuencias negativas de tipo estatal, académico, social, político y militar (Boudali *et al.*, 2022; Kumar y Kumar Sharma, 2022).

Características

De acuerdo con Lucena López (2016) y Ortega Triguero *et al.* (2006), un criptosistema se define como una quintupla (M, C, K, E, D) , donde:

M : constituye el conjunto de los mensajes originales que se desean enviar, también conocidos como *Plaintext*, los cuales aún no han sido cifrados.

C : representa el conjunto de los mensajes una vez han sido cifrados, también conocidos como criptogramas.

K : determina el conjunto de claves que se pueden utilizar en el criptosistema.

E : está formada por las transformaciones definidas de M a C . Para cada una de las claves posibles del conjunto K , va a existir una transformación diferente E_k .

D : hace referencia al conjunto de funciones o transformaciones (similar al conjunto E), que se aplican a cada elemento del conjunto C para obtener el mensaje original.

A partir de esto, se deduce que todo criptosistema debe cumplir las expresiones (136).

$$\begin{aligned} E_k(m_i) &= c_i \\ D_k(c_i) &= m_i \end{aligned} \tag{136}$$

Lo anterior significa que, al cifrar un mensaje m_i con la clave k , se obtiene el mensaje cifrado c_i . Si este mensaje se descifra con la misma clave k , se obtendrá el mensaje original.

En resumen, se debe cumplir con la condición dada en la expresión (137).

$$D_K(E_K(M)) = M \quad (137)$$

Entonces, la seguridad en un criptosistema depende del número y de la complejidad de las funciones o transformaciones aplicadas al mensaje, así como del tipo de clave que se emplee.

Tipos de criptografía

Los siguientes trabajos sentaron las bases para desarrollar la teoría de la información y la criptografía moderna: *A Mathematical Theory of Communication* y *Communication Theory of Secrecy Systems* de Claude Shannon (1948; 1949), y *New Directions in Cryptography* de Whitfield Diffie y Martín Hellman (1976). En esta última publicación, se introdujo la estrategia de criptografía asimétrica, la cual posibilitó que se generaran nuevos desarrollos con este énfasis.

Es conveniente precisar que la criptografía por sí sola hace referencia únicamente a la utilización de códigos para hacer ininteligible una información, pero no abarca las técnicas utilizadas para tratar de romper esos códigos, lo cual se enmarca dentro de lo que se conoce como criptoanálisis. Esto implica que las dos disciplinas de criptografía y criptoanálisis se relacionan directamente, lo que a su vez conlleva a que, siempre que se diseñe un mecanismo para encriptar cierta información, se debe tener presente cómo sería el criptoanálisis correspondiente, con el fin de evitar resultados inesperados (Lucena López, 2016).

El descubrimiento de un algoritmo de encriptación no es considerado como criptoanálisis, ya que el conocimiento de los algoritmos de cifrado no implica que pueda descifrarse el mensaje. Una forma de llevar a cabo el criptoanálisis es aplicar el algoritmo de descifrado a un mensaje encriptado

con cada una de las claves posibles y verificar cuáles de las salidas tienen más coherencia con un eventual texto plano. Esta técnica se conoce como fuerza bruta. Existen otros mecanismos de criptoanálisis que buscan aprovechar las debilidades detectadas en el algoritmo de encriptación, como el análisis diferencial.

Para hacer referencia tanto a la criptografía como al criptoanálisis, algunos autores han acudido al término criptología. Cabe resaltar que el hecho de que en la literatura exista una gran variedad de sistemas criptográficos conlleva que se despliegue una gama de técnicas de criptoanálisis. Esto está relacionado con el avance de la capacidad de los computadores, es decir que nuevos métodos de seguridad implican nuevos métodos de criptoanálisis. Nótese que la seguridad de un criptosistema se suele medir en términos del tiempo necesario para romper el mecanismo de cifrado y en función del dinero requerido para este propósito con garantías de éxito (Lucena López, 2016).

Los criptosistemas se consideran seguros si cumplen estos tres requerimientos: integridad, confidencialidad y disponibilidad de la información. Esto indica que garantizar la seguridad de un criptosistema sea un objetivo quizá de mayor nivel que el simple hecho de encriptar la información en sí. Además, se clasifican en dos tipos dependiendo de si se usa o no la misma clave para cifrar y descifrar (Lucena López, 2016; Ramió Aguirre, 2018).

Se puede hacer otro tipo de clasificación de los sistemas criptográficos de acuerdo con la forma en que se trate la información original en el mecanismo de cifrado, diferenciando entre sistemas de cifrado en flujo y sistemas de cifrado en bloque. Para el cifrado en flujo, se debe contar con una secuencia de claves, ya que por cada bit o byte a encriptar se utiliza una clave diferente. En el caso de cifrado en bloques, la información original se divide en grupos de bytes, los cuales se encriptan con la misma clave (Ramió Aguirre, 2018).

Criptografía simétrica

La criptografía simétrica, también conocida como clave privada, utiliza la misma clave para cifrar y descifrar. Lo anterior exige que el canal de intercambio de esta entre el emisor y el receptor sea seguro para que un intruso no logre interceptarla, ya que, en caso contrario, la seguridad del esquema

de cifrado se rompe totalmente. Para profundizar en este tema, se puede consultar a Lucena López (2016) y Ramió Aguirre (2018).

Algunos algoritmos que se destacan en esta dirección son: *Data Encryption Standard*, *International Data Encryption Algorithm* y *Advanced Encryption Standard*.

Data Encryption Standard (DES) (1976): fue desarrollado por la Agencia Nacional de Seguridad (*National Security Agency* [NSA]) por petición de Estados Unidos, bajo presión del sector empresarial para la protección de sus comunicaciones. Aunque durante muchos años fue utilizado como un método estándar de protección, en la actualidad, es catalogado como inseguro porque utiliza una clave de 56 bits, que resulta débil frente al actual poder computacional.

DES es un algoritmo de cifrado por bloques de 64 bits y utiliza claves del mismo tamaño. Sin embargo, en realidad, la clave resulta de 56 bits, ya que emplea 8 bits para comprobar la paridad.

En los algoritmos simétricos de cifrado por bloques, se divide la información que se desea encriptar en grupos del mismo tamaño y sobre cada uno se implementan las fases de permutación y difusión. La permutación consiste en tratar de ocultar la relación existente entre la información original, la información encriptada y la clave; mientras que con la difusión se busca distribuir la influencia de cada bit del mensaje original a lo largo del mensaje cifrado (Hernández Encinas, 2016).

El algoritmo DES se basa en otro algoritmo conocido en la literatura como LUCIFER, desarrollado por IBM a principios de la década de 1970. En 1998, se demostró que era posible *romper* el algoritmo DES, utilizando un ataque por fuerza bruta, debido a la longitud de la clave que usa. Sin embargo, dicho algoritmo no ha demostrado ninguna debilidad considerable. En 1998, la empresa *Electronic Frontier Foundation* (EFF) logró diseñar una máquina capaz de descifrar un mensaje DES en menos de tres días. A pesar de estos hallazgos, DES sigue siendo utilizado en una gran variedad de aplicaciones, como las transacciones en cajeros automáticos. Se han suscitado algunas variantes de DES conocidas como DES múltiple, DES con subclaves independientes y DES generalizado; todas estas modificaciones buscan ampliar el tamaño de la clave (Lucena López, 2016).

International Data Encryption Algorithm (IDEA) (1992): es un algoritmo considerado en la actualidad como muy seguro, toma bloques de 64 bits de longitud y una clave de 128 bits. Hasta el momento, ha mostrado ser resistente a varios ataques dentro de los que se destaca el criptoanálisis diferencial. La longitud de la clave hace que en la práctica sea imposible un ataque por fuerza bruta. IDEA implementa las fases de difusión y confusión por medio de operaciones XOR, sumas y productos módulo (Hernández Encinas, 2016).

Advanced Encryption Standard (AES) (1997): fue la propuesta ganadora en un concurso convocado por el Instituto Nacional de Normas y Tecnología (NIST), que tuvo gran acogida y pasó a ser un método estándar avalado por la *Federal Information Processing Standard (FIPS)* en el 2002. Como estaba enfocado a aplicaciones criptográficas no militares, se convertiría en el sucesor de DES. En la actualidad, es el algoritmo más popular en criptografía simétrica.

AES es un algoritmo de cifrado en bloques, apto para manejar longitudes de bloque y clave variables en rangos entre 128 y 256 bits. Estados Unidos lo acogió como estándar, utilizando claves de 128, 192 o 256 bits, y bloques de un tamaño de 128 bits.

Aunque AES involucra estrategias de permutación y difusión, es fácil y rápido de implementar tanto en *software* como en *hardware*. Además, requiere poca memoria debido a que usa transformaciones sencillas como operaciones XOR y funciones de desplazar filas o mezclar columnas; en todo caso, este se interpreta dentro de la estructura del cuerpo de Galois, formado por los enteros módulo 2^8 , denotado por $GF(2^8)$. Según sus autores, es poco probable que existan claves débiles, es decir, que puedan ser descubiertas por un atacante, ya que el diseño de las claves busca eliminar todo tipo de simetrías, además, es resistente a ataques diferenciales. Este método es considerado altamente seguro hasta la fecha (Lucena López, 2016).

Criptografía asimétrica

La criptografía asimétrica, también conocida como clave pública, emplea dos claves: una de dominio público y otra privada, las cuales se usan para cifrar y descifrar, respectivamente. Los algoritmos asimétricos han sido una buena alternativa para ser empleados en redes de comunicación inseguras.

De manera general, se basan en plantear al atacante problemas matemáticos difíciles de resolver. Para profundizar en este tema, se puede consultar a Hernández Encinas (2016) y Lucena López (2016).

La criptografía asimétrica tiene como desventaja que el mensaje cifrado es de mayor tamaño comparado con el original. Esto hace que en muchos esquemas de cifrado se utilice un modelo simétrico y que, para el mecanismo de distribución de clave, se recurra a un modelo asimétrico a fin de que sea más rápido y menos costoso.

Si el proceso de generación de las claves fuese de alguna forma predecible, esto conllevaría que un intruso podría llegar a adivinar la siguiente clave a partir de una o varias de ellas. Por esto, es indispensable que, en un algoritmo criptográfico, no exista ningún tipo de dependencia entre una clave y la siguiente, es decir que deben ser aleatorias. Esto hace que la generación de números aleatorios sea de gran interés en el área de la criptografía.

Para algoritmos de tipo simétrico, una clave de 128 bits es considerada segura; mientras que, para algoritmos asimétricos, se recomienda que el tamaño de la clave sea de al menos 2048 bits, exceptuando los algoritmos basados en curvas elípticas. Se destacan dos aplicaciones de la criptografía simétrica: una se refiere al cifrado de la información sin necesidad de transmitir la clave de decodificación y la otra tiene que ver con la autenticación de mensajes (Ibarra Quevedo *et al.*, 2001).

En este sentido, se resaltan los siguientes algoritmos (Hernández Encinas, 2016; Lucena López, 2016):

RSA (1977): llamado así en honor a sus creadores Rivest, Shamir y Adleman, este algoritmo aprovecha la dificultad que existe para factorizar números primos de gran tamaño, escoge de forma aleatoria dos números primos grandes y los mantiene en secreto. Con estos, construye las claves pública y privada. Este algoritmo ha sobrepasado una multitud de ataques, ya que la longitud de la clave es bastante extensa. Además, es fácil de implementar y es considerado a la fecha uno de los más seguros, dado que un atacante se enfrenta a los problemas de factorización o de logaritmo discreto, que son computacionalmente intratables.

Diffie Hellman (1976): llamado así en honor a sus creadores, fundamentalmente, es un sistema para intercambio de claves a través de un canal inseguro y sin autenticación. Es aplicable para intercambiar claves simétricas de manera segura. La ventaja de este sistema es que no se requieren en sentido estricto claves públicas, sino una información compartida por los dos interlocutores.

ElGamal: inicialmente, fue diseñado para firmas digitales; más tarde, se extendió para encriptar mensajes. Su base conceptual es el problema de los logaritmos discretos ligado al problema de factorización, junto con el algoritmo de Diffie-Hellman.

Digital Signature Algorithm (DSA) (1991): este algoritmo, propuesto por el NIST, es una modificación del método de ElGamal y está dirigido a la autenticación de firmas digitales. Selecciona un número primo de gran tamaño, a partir del cual genera las claves privada y pública mediante operaciones módulo.

En el campo de la criptografía asimétrica, se han generado varios algoritmos utilizando curvas elípticas. Esto ha sido posible gracias a que presentan una serie de propiedades que originan problemas análogos a los de la aritmética modular. Los primeros desarrollos en criptografía basada en curvas elípticas fueron planteados por Neal Koblitz y Victor Miller en 1985 (Blake *et al.*, 1999; Shaikh *et al.*, 2017; Villanueva Polanco, 2014).

A pesar de que la estructura algebraica de las curvas elípticas es abstracta, su implementación resulta ser más cómoda comparada con los métodos de criptografía basados en aritmética modular. En el contexto de curvas elípticas, existe un problema análogo al del logaritmo discreto en grupos finitos de enteros y aún no se ha encontrado un algoritmo eficiente para resolverlo. Por tanto, esta situación ha sido aprovechada con éxito para la implementación de estrategias criptográficas de clave pública. Una ventaja es que, utilizando claves pequeñas, el nivel de seguridad de estos esquemas es alto.

Debido a la similitud entre el problema del logaritmo discreto en curvas elípticas y el problema del logaritmo complejo en grupos finitos de enteros, muchos de los algoritmos asimétricos que se apoyan en el problema de la

factorización de números primos han sido replanteados utilizando el lenguaje de curvas elípticas (Gayoso Martínez *et al.*, 2018).

Los modelos criptográficos antes descritos son ineficientes cuando se desea proteger información de gran tamaño, como imágenes, videos o audios. Una alternativa, abordada en el capítulo 6 y en la cual se enfoca este libro, corresponde al uso de atractores caóticos para generar secuencias pseudoaleatorias y emplearlas en las fases de difusión y permutación del proceso de encriptación. En la figura 30, se muestra un ejemplo del uso de los atractores de Lorenz y Rössler junto con la teoría de sincronismo para definir un modelo criptográfico aplicado a textos (Moreno *et al.*, 2016).

Figura 30. Encriptación de texto usando los atractores caóticos de Lorenz y Rössler

Mensaje a encriptar	La revista Ingeniería es una publicación periódica de la Facultad de Ingeniería de la Universidad Distrital Francisco José de Caldas.
Mensaje encriptado usando el atractor de Lorenz	g ;?jĒR@Y±-Ee.Ā ±¼¿%ÄERÖÜðàØTÖÖIĒĒÇ~C. s~ª.«°RJqçl==~>SKqH\$§j°~c»- %μ%İ¿İ°ÖÜæÐÖ`İÖİçÖ
Mensaje encriptado usando el atractor de Rössler	q.vĒ»İ¿ĒĒ.vÄ%»Ä¿»ĒN.v»ĒvĒÄ.vÆĒ.Ä¿¹.¿çhÄvÆ»Ēhª¿¹.vª»v Ä.v.¹ĒÄĒ.ªvª»vÄ%»Ä¿»ĒN.vª»vÄ.vª»vÄ¿İ»ĒĒ¿ª.ªv¿ĒĒÇ%ĒĒAuÇġ Ä.¿Ē.ÄuÄĒİu²ªuġÄ²ġĒ

Fuente: Moreno *et al.* (2016).

Referencias

- Bingöl, S. (2022). Methods for Encryption in Early 19th-Century Ottoman Diplomatic Correspondence. *Cryptologia*, 46(6), 498-524.
<https://doi.org/10.1080/01611194.2021.1919943>
- Blake, I. F., Seroussi, G. y Smart, N. (1999). *Elliptic Curves in Cryptography*. Cambridge University Press.
- Boudali, A., Said, N. H. y Ali-Pacha, A. (2022). A New Symmetrical Cryptosystem Based Cellular Automata and Chaotic Map Function. *Journal of Discrete Mathematical Sciences and Cryptography*, 25(5), 1435-1455. <https://doi.org/10.1080/09720529.2021.2020427>
- Diffie, W. y Hellman, M. (1976). New Directions in Cryptography. *IEEE Transactions on Information Theory*, 22(6), 644-654.
<https://doi.org/10.1109/TIT.1976.1055638>

- Gayoso Martínez, V., Hernández Encinas, L. y Martín Muñoz, A. (2018). *Criptografía con curvas elípticas*. Consejo Superior de Investigaciones Científicas.
- Hernández Encinas, L. (2016). *Qué sabemos de la criptografía*. Catarata.
- Ibarra Quevedo, R., Serrano López, M. A. y García y González, C. C. (2001). *Teoría de la información y encriptamiento de datos*. Instituto Politécnico Nacional.
- Kumar, A. y Kumar Sharma, S. (2022). Information Cryptography Using Cellular Automata and Digital Image Processing. *Journal of Discrete Mathematical Sciences and Cryptography*, 25(4), 1105-1111. <https://doi.org/10.1080/09720529.2022.2072437>
- Lucena López, M. J. (2016). *Criptografía y seguridad en computadores*.
- Moreno, J., Parra, F., Huérfano, R., Suárez, C. y Amaya, I. (2016). Modelo de encriptación simétrica basada en atractores caóticos. *Ingeniería*, 21(3), 378-390. <http://www.scielo.org.co/pdf/inge/v21n3/v21n3a08.pdf>
- Ortega Triguero, J. J., García del Castillo Crespo, E. C. y López Guerrero, M. A. (2006). *Introducción a la criptografía: historia y actualidad*. Ediciones de la Universidad de Castilla-La Mancha.
- Ponemon Institute LLC, & Accenture Security. (2019). *The Cost of Cybercrime. Ninth Annual Cost of Cybercrime Study. Unlocking the Value of Improved Cybersecurity Protection*. Accenture. <https://www.informeticplus.com/the-cost-of-cybercrime-ninth-annual-cost-of-cybercrime-study-unlocking-the-value-of-improved-cybersecurity-protection-accenture-security-y-ponemon-institute>
- Ramió Aguirre, J. (2018). *Curso de criptografía aplicada*. Universidad Politécnica de Madrid.
- Shaikh, J. R., Nenova, M., Iliev, G. y Valkova-Jarvis, Z. (2017). Analysis of Standard Elliptic Curves for the Implementation of Elliptic Curve Cryptography in Resource-Constrained E-Commerce Applications. *2017 IEEE International Conference on Microwaves, Antennas, Communications and Electronic Systems (COMCAS)*, Tel-Aviv, Israel. <https://doi.org/10.1109/COMCAS.2017.8244805>
- Shannon, C. E. (1948). A Mathematical Theory of Communication. *The Bell System Technical Journal*, 27(3), 379-423. <https://doi.org/10.1002/j.1538-7305.1948.tb01338.x>

- Shannon, C. E. (1949). Communication Theory of Secrecy Systems. *The Bell System Technical Journal*, 28(4), 656-715. <https://doi.org/10.1002/j.1538-7305.1949.tb00928.x>
- Villanueva Polanco, R. (2014). Algoritmos básicos para la multiplicación de puntos en una curva elíptica. *Investigación e Innovación en Ingenierías*, 2(1), 49-55. <https://doi.org/10.17081/invinno.2.1.2057>

Capítulo 6. Nuevas propuestas criptográficas

Reseña histórica

En consonancia con los avances tecnológicos de las últimas décadas y dada la necesidad de intercambiar una gran cantidad de información por las redes de comunicación de forma segura, paralelamente en las últimas décadas, han surgido nuevos esquemas de encriptación utilizando principios matemáticos como teoría del caos, autómatas celulares, cuaterniones, código ADN y algoritmos genéticos. Esto ha llevado a consolidar comunidades académicas especializadas en alguna de estas líneas y ha producido un número significativo de trabajos publicados en revistas científicas con resultados prometedores ante la necesidad de encriptar información de gran tamaño, como imágenes tanto en escala de grises como a color, videos o audios, mostrando que estos enfoques arrojan resultados cercanos a lo esperado y cumplen con las condiciones requeridas de privacidad, integridad, autenticación y no repudio. Por lo general, se han obtenido resultados satisfactorios en la etapa de evaluación de indicadores de desempeño, lo cual se ha constituido en un reto constante para las comunidades científicas interesadas en la seguridad de la información, con altos estándares de seguridad y desempeño. Para profundizar en este tema, se puede consultar a Al-Mashhadi y Abduljaleel (2017), Bakhshandeh y Eslami (2013), Boudali *et al.* (2022), Jose y Chowdhury (2017), Kumar y Kumar Sharma (2022), Mendez *et al.* (2023) y Nagaraj *et al.* (2015).

Con el propósito de contribuir desde la academia a la formulación de propuestas novedosas de criptografía, utilizando nuevas herramientas matemáticas y de inteligencia artificial, el grupo de complejidad de la Universidad Distrital ComplexUD desarrolló un proyecto de investigación titulado *Modelos de encriptación basados en atractores caóticos*. Este fue formalizado ante la Oficina de Investigaciones de la Universidad. Como resultado de este proyecto, se presenta la síntesis de algunos de los trabajos que dieron origen a proyectos de grado o artículos de investigación.

Motivación

Este proyecto estuvo motivado por la necesidad de que, desde las instituciones de educación superior, se promueva el interés por trabajar en el diseño y consolidación de nuevos esquemas de seguridad, utilizando elementos matemáticos y de inteligencia artificial para afrontar los desafíos de las nuevas formas de comunicación masiva, exacerbados por la pandemia que ocasionó la COVID-19 entre el 2019 y el 2021. Además, los métodos convencionales de criptografía no son idóneos para la protección de información de gran tamaño como imágenes, audios o videos, principalmente, por el gran volumen de información que manejan y por la redundancia que pueden llegar a presentar.

Los modelos criptográficos encontrados en la literatura científica sobre el uso de la teoría del caos aprovechan las propiedades inherentes a este tipo de comportamiento para generar secuencias pseudoaleatorias, que son usadas en la definición del algoritmo de encriptación. Se utilizan sistemas dinámicos caóticos de una o más dimensiones o hasta de dimensión fraccionaria, a fin de obtener mayor seguridad y mejores indicadores de desempeño. Este enfoque de trabajo ha hecho que se propongan y se estudien nuevos sistemas caóticos para utilizarlos con el mismo propósito, considerando que, entre mayor sea la complejidad del atractor caótico a usar en el cifrado, mejores serán los resultados arrojados en el sistema de criptografía.

La importancia de utilizar sistemas dinámicos caóticos para generar estrategias de cifrado radica en la similitud que existe entre las propiedades comportamentales de un sistema caótico, como ergodicidad, propiedades de mezcla, dependencia sensitiva de los parámetros y condiciones iniciales, y

las propiedades ideales de sistemas de encriptación para imágenes, audios o videos, basados en los procesos de difusión y permutación (Wu *et al.*, 2017).

A continuación, se describen algunos trabajos que fueron desarrollados durante la ejecución del proyecto, al cual se vincularon estudiantes del programa de Ingeniería de Sistemas interesados en estos ejes temáticos. En este sentido, se consolidaron propuestas de criptosistemas, enfocadas a imágenes en color, videos transmitidos en tiempo real, audios y textos complementados con esteganografía.

Aplicación de atractores caóticos en imágenes a color

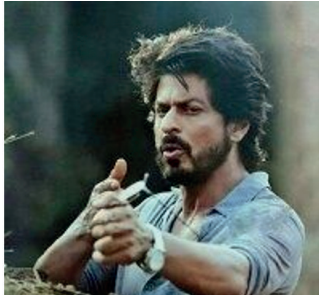
El propósito central consistió en diseñar un algoritmo para encriptar imágenes en escala de colores RGB por medio de la utilización de sistemas dinámicos caóticos. Para este fin, se tuvieron en cuenta algunos referentes de la literatura científica sobre criptografía y caos aplicables a información de gran tamaño, con el objetivo de definir criterios propios para esquematizar una propuesta original (Santos Bustos *et al.*, 2020), cuyos indicadores de seguridad y desempeño se enmarcaran dentro de los estándares reportados en la literatura.

Síntesis del modelo propuesto

Inicialmente, se exploraron algunos sistemas dinámicos caóticos y se decidió implementar la fase de permutación por medio del algoritmo de línea de onda, el cual es definido a partir del sistema caótico invertible Cat de Arnold.

Dado que, para la fase de difusión, se obtenían buenos resultados tanto con el atractor hipercaótico de Chen como con el atractor hipercaótico de Lorenz, se decidió trabajar el algoritmo con la posibilidad de seleccionar cualquiera de estos dos atractores. Los sistemas hipercaótico de Chen e hipercaótico de Lorenz para ciertos valores de los parámetros en consideración tienen un comportamiento dinámico complejo, por lo que se consideraron apropiados para definir el mecanismo de difusión de píxeles. En la figura 31, se muestra el resultado de difundir una imagen permutada tanto con Chen como con Lorenz, lo cual resulta satisfactorio para el ojo humano.

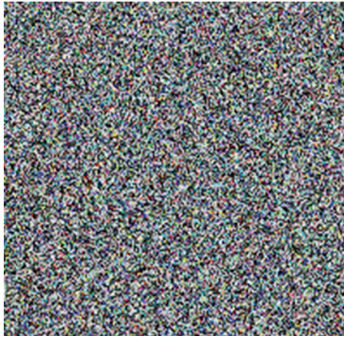
Figura 31. Ilustración de etapa de difusión, utilizando Chen y Lorenz



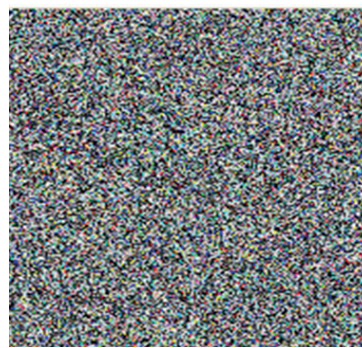
(a) Imagen original



(b) Imagen permutada
con tres iteraciones



(c) Imagen difundida,
utilizando Chen



(d) Imagen difundida,
utilizando Lorenz

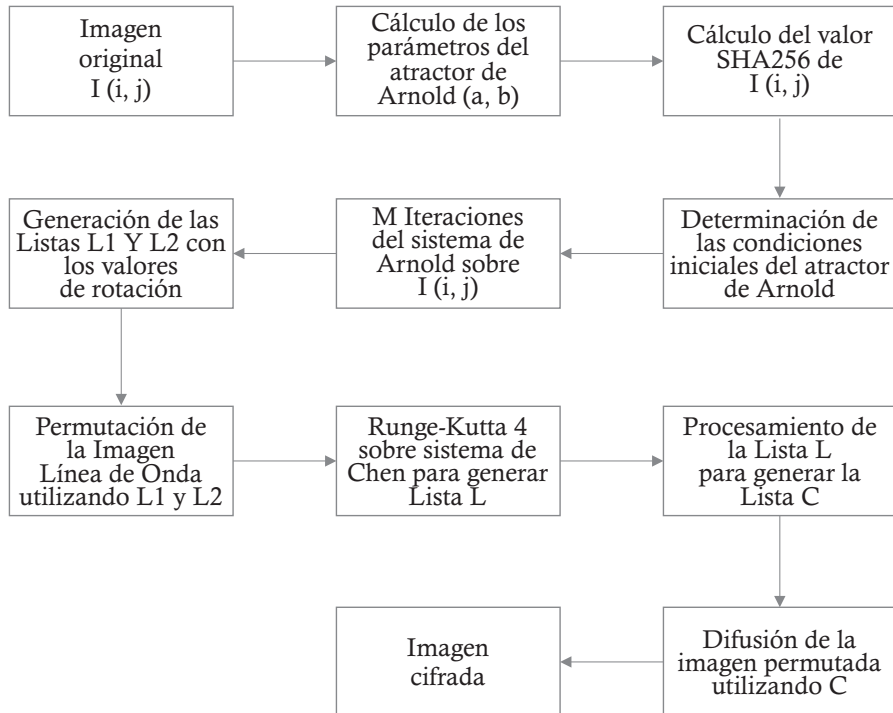
Fuente: Santos Bustos *et al.* (2020).

En la fase de permutación, se suman todos los valores de los píxeles de una capa RGB y se consigue una variable que se utiliza para definir los parámetros del atractor Cat de Arnold. Simultáneamente, se calcula el SHA 256 de la imagen a color. A partir de esto, se calculan las condiciones iniciales del atractor Cat de Arnold, se itera y los valores obtenidos se emplean en la definición de las rotaciones circulares llevadas a cabo por medio del procedimiento de línea de onda.

Al utilizar uno de los atractores hipercaóticos de Chen o Lorenz, los cuales son resueltos numéricamente por medio de Runge Kutta 4, y la operación XOR, se define finalmente el proceso de permutación.

En la figura 32, se sintetiza el modelo desarrollado que fue publicado en la revista *Ingeniería* de la Universidad Distrital en el 2020 (Santos Bustos *et al.*, 2020).

Figura 32. Esquema del algoritmo de encriptación para imágenes a color



Fuente: Santos Bustos *et al.* (2020).

Para validar la propuesta original, se aplicaron pruebas de seguridad y desempeño.

Evaluación del desempeño

Lo anterior implicó la obtención de indicadores de seguridad y desempeño correspondientes al uso de cada atractor. Este hecho se consideró como una ventaja porque se compararon los resultados obtenidos al interior de la propuesta y, además, se tomaron en cuenta algunos de los trabajos consultados para realizar la correspondiente comparación de indicadores. Se destaca que, en ambos casos, los resultados arrojados por la propuesta propia se ajustaban dentro de los ideales de esquemas criptográficos seguros.

La metodología implementada para definir el algoritmo, desarrollado en lenguaje C++, implicó su aplicación por separado en cada uno de los canales RGB. Se destaca que, al buscar reducir los tiempos de ejecución, se empleó procesamiento en paralelo. Utilizando la librería POSIX de Linux `pthread.lib` que permite la implementación de hilos y empleando cuatro de estos, se logró reducir cerca de un 17% el tiempo de ejecución. Respecto al tamaño de espacio de clave, esta propuesta indica que es de 2^{641} , cifra considerable que ayuda a reafirmar la seguridad del criptosistema propuesto, junto con los valores de correlación, entropía, NPCR (Number of Changing Pixel Rate) y UACI (Unified Averaged Changed Intensity). Asimismo, se abordó la propiedad de sensibilidad de la clave.

En síntesis, con esta propuesta se buscó, por medio de un ejercicio académico, incentivar a que otros expertos se interesen por el tema y a que se unan esfuerzos en pro de articular los sistemas dinámicos caóticos con la criptografía, a fin de crear esquemas de encriptación que respondan a los desafíos impuestos a través del avance y desarrollo de los sistemas de intercambio de información.

Aplicación de atractores caóticos en videos transmitidos en tiempo real

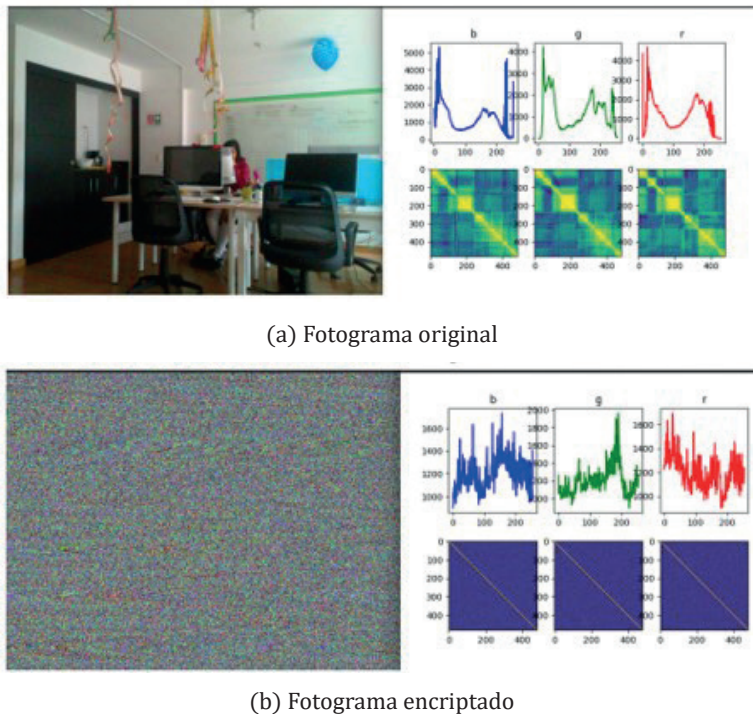
Se planteó un algoritmo para el cifrado de videos en tiempo real, que se conoce como *streaming*. Este dio origen a un trabajo de grado de un estudiante del programa de Ingeniería de Sistemas (Barragán Guzmán y Arias Sarria, 2019).

Síntesis del modelo propuesto

Mediante la combinación de técnicas de permutación y difusión, y utilizando componentes de *software* dispuestos para el manejo de videos, se consolidó un algoritmo para encriptar videos en tiempo real, aprovechando las características comportamentales del sistema dinámico de Lorenz enmarcadas dentro del caos y la sincronización. En la figura 33, se muestra el resultado del proceso de encriptación de un fotograma, junto con los diagramas de correlación de la imagen original y de la salida encriptada.

La sincronización surgió al estudiar la presencia de oscilaciones caóticas. Pecora y Carroll (1990) demostraron experimentalmente que puede haber comportamientos caóticos con apariencia imprevisible y que, a pesar de tener condiciones iniciales diferentes y cercanas, se pueden confundir a largo plazo en una sola trayectoria. El atractor caótico de Lorenz tiene esta propiedad.

Figura 33. Gráficos de correlación

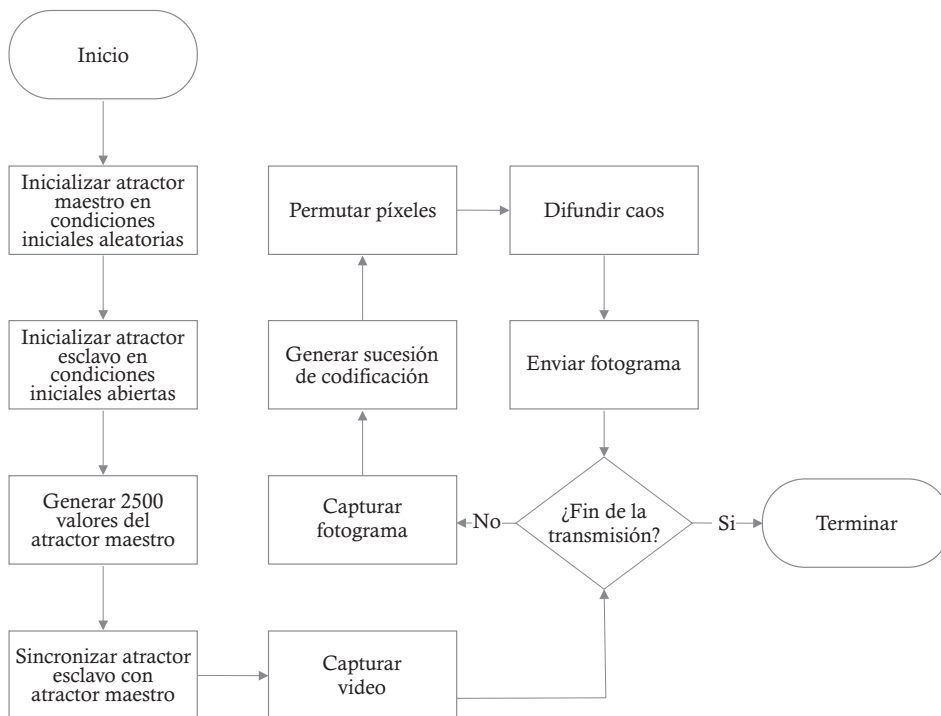


Fuente: Barragán Guzmán y Arias Sarria (2019).

El modelo planteado en este trabajo presenta una limitación que consiste en que los videos a encriptar no tienen audio, ya que el algoritmo encripta únicamente los fotogramas de tamaño $m \times n$, los cuales se muestrean cada 0.03 segundos. Por medio de Runge Kutta 4 aplicado sobre el atractor de Lorenz, se consigue una sucesión de longitud $m \times n$, comenzando en la

iteración 2501; una parte de las entradas de esta se utilizan para la fase de permutación y las demás se emplean para el esquema de difusión de píxeles. En la figura 34, se muestran esquemáticamente las fases llevadas a cabo para implementar el modelo propuesto.

Figura 34. Esquema del algoritmo de encriptación de videos



Fuente: Barragán Guzmán y Arias Sarria (2019).

Para validar el trabajo propuesto, se llevaron a cabo pruebas de evaluación y desempeño.

Evaluación del desempeño

En este apartado, se realizaron diferentes pruebas utilizando el mismo computador. Se cifraron videos obtenidos directamente desde la cámara web y se simuló el envío. Se observó que los fotogramas de una parte se encriptaban y los de otra, mediante la aplicación de las mismas estrategias de cifrado, pero

en sentido inverso, permitían recuperar el contenido multimedia sin pérdida de información.

Los resultados encontrados basados en las pruebas efectuadas, como resistencia a ataques diferenciales, cálculo de valores de entropía, valores de correlación de píxeles adyacentes y análisis de sensibilidad de la clave, permitieron enmarcar la propuesta como un esquema de seguridad robusto en la transmisión de videos en tiempo real. Asimismo, los tiempos requeridos para encriptar y desencriptar son mínimos; esto permite que, en el momento de recuperar la información, no se presenten retrasos en la renderización.

Aplicación de la teoría de cuaterniones

En los reportes de la literatura científica, recientemente, se han dado a conocer trabajos de encriptación que utilizan la teoría de cuaterniones. De igual manera, son varios los modelos de encriptación publicados en las últimas décadas orientados a archivos multimedia. En este sentido, a partir del trabajo de Khalil (2017), se decidió consolidar una propuesta de sistema de encriptación que fuera aplicable a archivos multimedia, utilizando cuaterniones y sistemas dinámicos caóticos, y cumpliendo estándares de seguridad. Esto dio origen a un trabajo de grado para optar por el título de Ingeniero de Sistemas en la Universidad Distrital (Méndez López y Ladino Torres, 2019).

Aprovechando la estructura matemática de los cuaterniones, se evidenciaron las ventajas que se tienen al momento de representar un archivo multimedia en estos, es decir, se cuenta con cuatro componentes: la primera destinada al audio y las otras tres a cada uno de los canales RGB. Por esta razón, se adoptaron como ambiente de trabajo para fusionar con la teoría de sistemas dinámicos caóticos, buscando consolidar un sistema de encriptación enfocado a archivos multimedia.

Síntesis del modelo propuesto

En esta propuesta, se tomó como base el atractor caótico Julia cuaternión, que funciona de forma análoga al conjunto de Julia complejo, pero, en lugar de utilizar números complejos, se emplean números cuaterniones, por lo que el resultado ya no es un fractal en dos dimensiones, sino en cuatro dimensiones. En este contexto, la clave del esquema de encriptación es una imagen

que se obtiene a partir de un plano de corte que se aplica sobre el sistema dinámico caótico Julia cuaternión. Esto resultó sencillo gracias a la utilización de la herramienta POV-Ray (The Persistence of Vision Raytracer), que consiste en un *software* de libre acceso, enfocado en la generación de gráficos en tres dimensiones y con la ventaja de que incluye una implementación nativa del fractal Julia cuaternión.

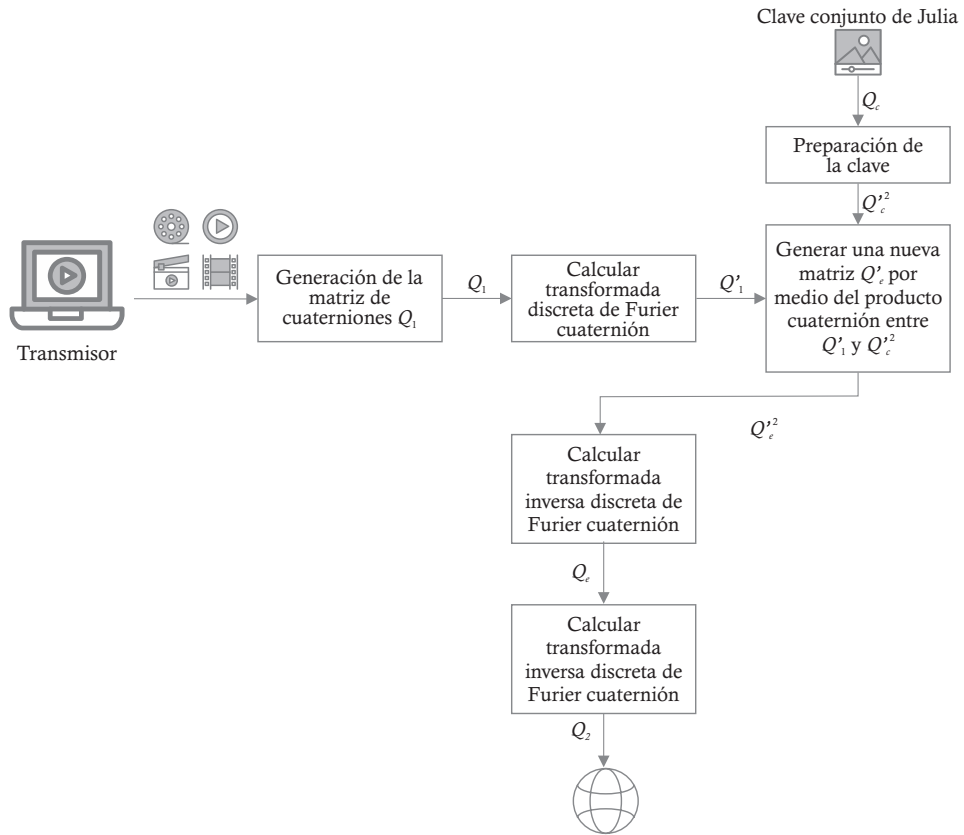
La figura 35 muestra uno de estos fractales con $c = 0 + 0.5i + 0_j + 0_k$, $pc = 0,12 + 1i + 1_j + 1_k$, $n = 200$, $RGB_1 = (1, 0.5, 0.2)$, $RGB_2 = (1, 0.5, 0.2)$, $RGB_3 = (1, 0.65, 0)$, $RGB_4 = (1, 0.65, 0)$, 1280×720 y $Fr = 30$. Para la distribución de la clave, se utilizó un esquema documentado en la literatura que viene a ser una modificación de Diffie-Hellman (Valluri y Narayan, 2016).

Figura 35. Fractal Julia cuaternión



Fuente: Méndez López y Ladino Torres (2019).

En el modelo de encriptación, se utilizó la transformada discreta de Fourier cuaternión, la cual se calculó empleando la librería QTFM de Matlab. Cabe resaltar que tanto el archivo multimedia como la imagen clave son denotados desde el inicio en el contexto de cuaterniones y que sobre esas matrices cuaternión se aplica la transformada discreta de Fourier cuaternión. En la figura 36, se sintetiza el proceso de encriptación llevado a cabo.

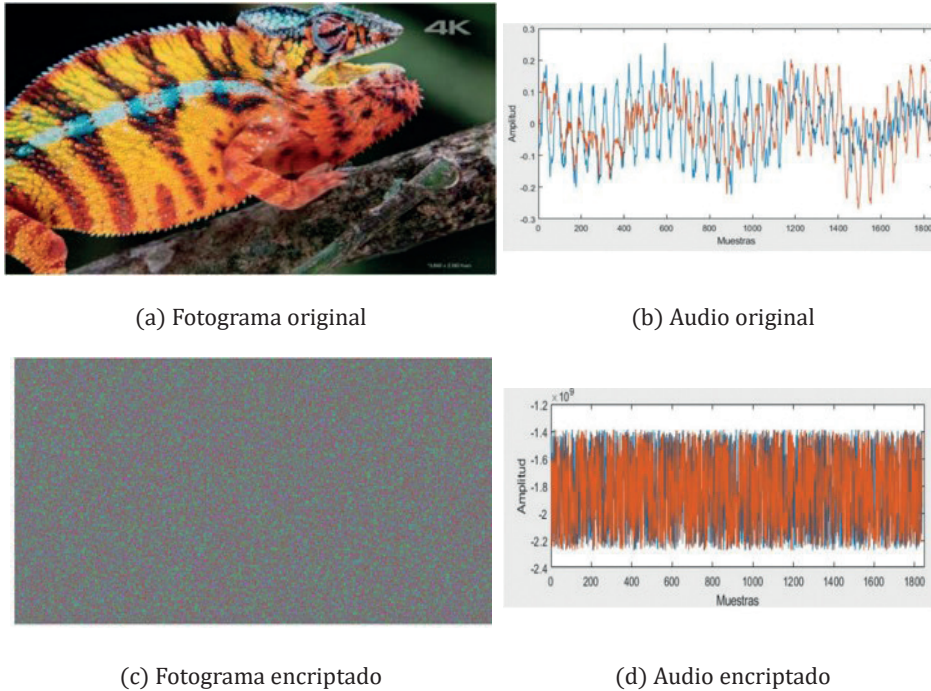
Figura 36. Proceso de encriptación de archivos multimedia

Fuente: Mendez *et al.* (2023).

En la figura 37, se ilustra el comportamiento del modelo propuesto.

Para conseguir una mejor sensibilidad del sistema, cuando el archivo a cifrar es una imagen, se descompone en los canales RGB y sobre cada capa se aplica una máscara por medio de la operación bitxor de Matlab con una matriz de números pseudoaleatorios, generados por una semilla compartida. Para recuperar el archivo multimedia original, se aprovecha la existencia de la transformada inversa discreta cuaternión y la operación de división de cuaterniones.

Figura 37. Resultado de encriptación de una muestra de archivo multimedia



Fuente: Méndez López y Ladino Torres (2019).

Evaluación del desempeño

Para garantizar la seguridad del sistema de encriptación propuesto, se llevaron a cabo varios análisis estadísticos que evidenciaron las siguientes fortalezas: el contenido multimedia encriptado presenta una distribución uniforme, los coeficientes de correlación tanto para la imagen como para el audio encriptados son próximos a cero y un pequeño cambio en el archivo original a encriptar incide notablemente en los resultados de la encriptación. Lo anterior permite asegurar que esta propuesta es segura ante ataques diferenciales.

Por otra parte, se determinó que la clave es altamente sensible a pequeñas perturbaciones. Dada la naturaleza del esquema propuesto, se consiguió que el espacio de clave fuera lo suficientemente amplio, lo cual provee seguridad frente a un posible ataque de fuerza bruta. También se hallaron los valores de entropía para imagen y para audio, los cuales se enmarcaron dentro de los valores esperados. Por consiguiente, no es pretencioso asegurar que las

pruebas presentadas en este trabajo permitieron enmarcar la propuesta dentro de estándares de seguridad recomendados en la literatura científica.

Finalmente, cabe mencionar que, por medio de este ejercicio académico, se lograron articular varios conceptos matemáticos, lo cual permitió, por un lado, su apropiación y, por otro, corroborar una vez más la importancia de las ciencias matemáticas en la formación ingenieril.

Aplicación de autómatas celulares

En este apartado, se presenta un modelo de encriptación simétrico que utiliza la teoría de autómatas celulares (AC) para proteger las regiones de interés en videos provenientes de cámaras de vigilancia. Este modelo evidencia indicadores de seguridad dentro de los estándares presentados en la literatura actual. Dicho trabajo dio origen a un artículo científico (Cortés Martínez *et al.*, 2020).

Esta propuesta aprovecha las características del comportamiento caótico en los AC para usarlas en la permutación y difusión del método criptográfico propuesto. Se toma como base el comportamiento original de los autómatas celulares unidimensionales, respetando el concepto de vecindad, teniendo en cuenta especialmente los autómatas de clase 3, ya que tienen comportamiento caótico, y centrándose de manera detallada en los autómatas reversibles, fundamentales a la hora de encriptar y desencriptar.

Síntesis del modelo propuesto

A continuación, se hace una descripción general del modelo propuesto, el cual utiliza autómatas celulares multicapa para representar las imágenes, reglas compuestas reversibles basadas en autómatas compuestos y extracción de regiones de interés sobre las imágenes. Estos elementos son fundamentales tanto para la formulación del criptosistema como para su evaluación.

Representación de imágenes como autómatas celulares multicapa: usualmente, las imágenes de las cámaras de vigilancia se capturan en escala de grises, lo cual es positivo porque se pretende garantizar que se mantenga la información de las imágenes y haya mayor robustez en el análisis. Cada uno de los fotogramas que constituyen el video puede verse como una matriz con

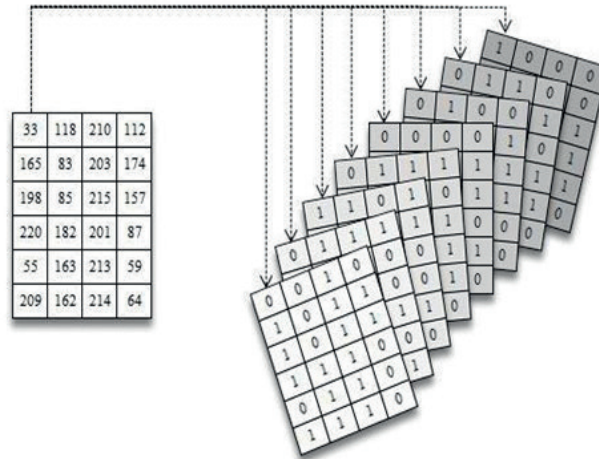
valores entre 0 y 255 (negro y blanco, respectivamente). El número 255 requiere de 8 bits (1 byte) para su representación en binario; por tanto, con el fin de unificar la equivalencia de todos los valores en la matriz, se toman 8 bits para denotar a cada uno de ellos, entonces, una imagen de $M \times N$ píxeles estará compuesta por $M \times N \times 8$ bits.

Después de convertir a binario los valores de los píxeles, se procede a ubicar cada bit en una matriz (o capa) independiente; de esta manera, se obtienen 8 capas para un fotograma. En la figura 38, se muestra la distribución de las capas para una imagen en escala de grises.

Una vez obtenidas las 8 capas para representar el fotograma, cada fila o columna de cada una de las capas se considera el estado inicial de un autómata celular unidimensional. De esta manera, se consigue un autómata celular multicapa.

Autómata celular compuesto: lo primero que se propone es un tipo especial de autómata celular unidimensional basado en los autómatas celulares elementales de Wolfram (AEs), donde se combinan las reglas de transformación de estos últimos. Por esta razón, se denominaron autómatas celulares compuestos. Con esas reglas surgen nuevos comportamientos que no se pueden obtener a partir de los AEs. Las reglas originales se aplican de manera tradicional con bordes periódicos, sin embargo, se van alternando para ser aplicadas en cada una de las celdas subsiguientes.

El modelo plantea la utilización de reglas compuestas reversibles (RCR), las cuales, como su nombre lo indica, solo contienen reglas elementales reversibles. Sin embargo, no todas las combinaciones generan reglas reversibles compuestas. Se debe tener en cuenta que las combinaciones contengan reglas con desplazamientos en la misma dirección o reglas de identidad con su complemento. Cada RCR se utiliza para el cifrado y su correspondiente complemento, para la recuperación de la información original. Las reglas se eligen con base en la entropía y la distancia de Hamming entre ellas.

Figura 38. Interpretación de una imagen como autómata celular multicapa

Fuente: Cortés Martínez *et al.* (2020).

Extracción de regiones de interés: el método de encriptación presentado se centra en las regiones del video donde se evidencia movimiento, denominadas regiones de interés (RoIs por sus siglas en inglés). Estas se obtienen al comparar cada fotograma del video con el inicial e identificar los pixeles que han cambiado. Si dichos pixeles están agrupados en un conjunto de 256 o más, esa región se considera un RoI y se encierra en un rectángulo. En caso de que dos RoIs se encuentren separados por menos de 16 pixeles, se juntan formando un único RoI. Con el fin de facilitar el proceso de aplicación de los autómatas celulares, la dimensión de cada RoI se ajusta a un número de pixeles múltiplo de 8.

A partir de un autómata celular multicapa, se extraen 1027 bits consecutivos con los que se realiza la operación XOR junto con las entradas de cada columna y se obtiene una secuencia de 1027 bits que constituirán la llave con la que se encriptan todos los RoIs del fotograma. Aquí se involucra la regla 30 para iterar sobre 1023 bits de la secuencia, los otros 4 bits indicarán

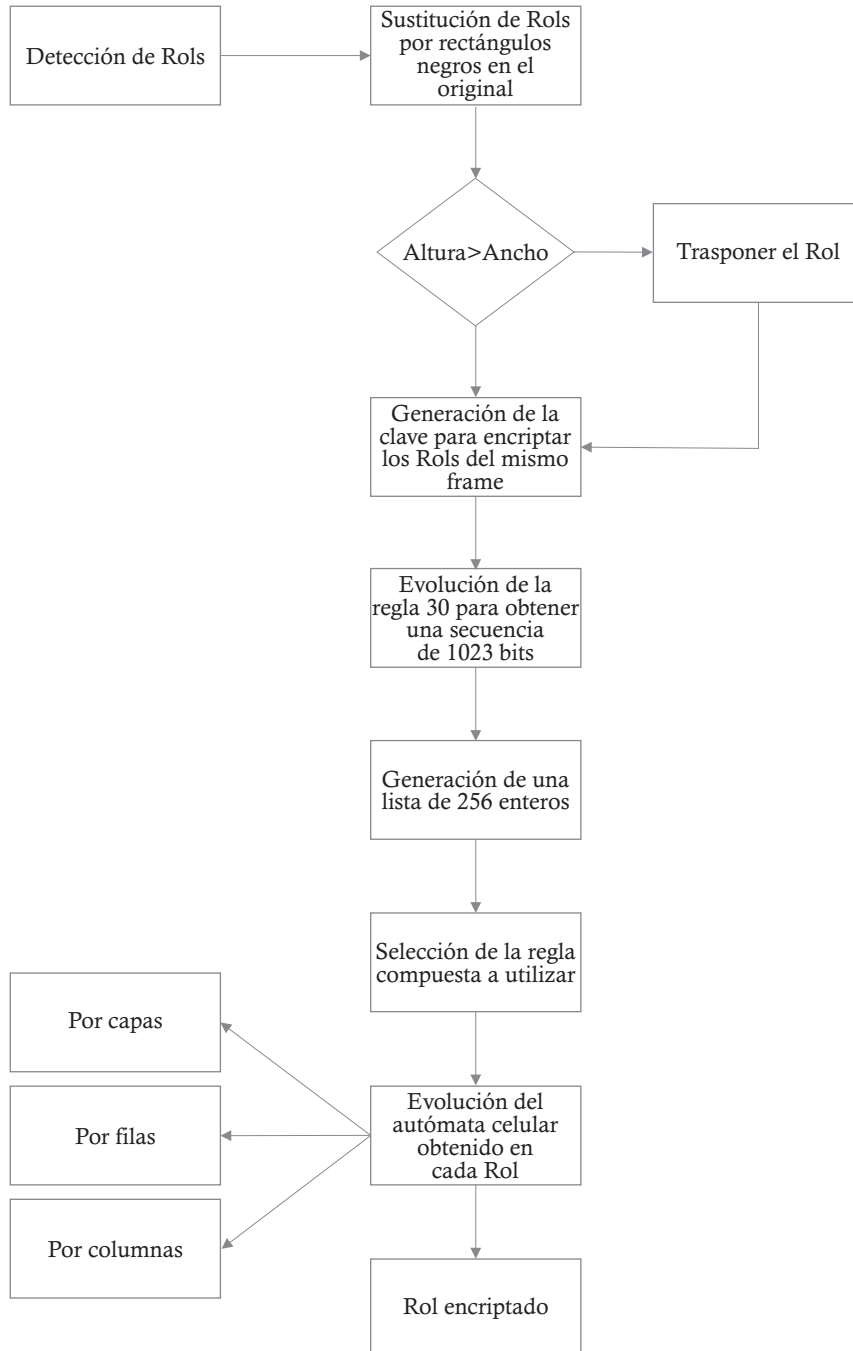
el número de iteraciones que se realizan. Asimismo, se determina el número de desplazamientos sobre la secuencia a fin de aumentar la aleatoriedad.

En resumen, el modelo de encriptación propuesto cifra solamente las regiones de interés reemplazándolas por rectángulos negros dentro del fotograma. Para realizar este cifrado, se representa cada RoI como un autómata celular multicapa. Uno de los RoIs del fotograma se escoge para generar la clave con la que se encriptarán los demás RoIs del mismo fotograma. Cada fotograma tendrá una clave diferente para cifrar sus RoIs.

La secuencia dinámica de bits generada permite cambiar el orden que se tomará como referencia, aplicándolo de arriba abajo, de izquierda a derecha y por columnas de fin a inicio. En la figura 39, se resume el proceso llevado a cabo para la encriptación de los RoIs de cada fotograma.

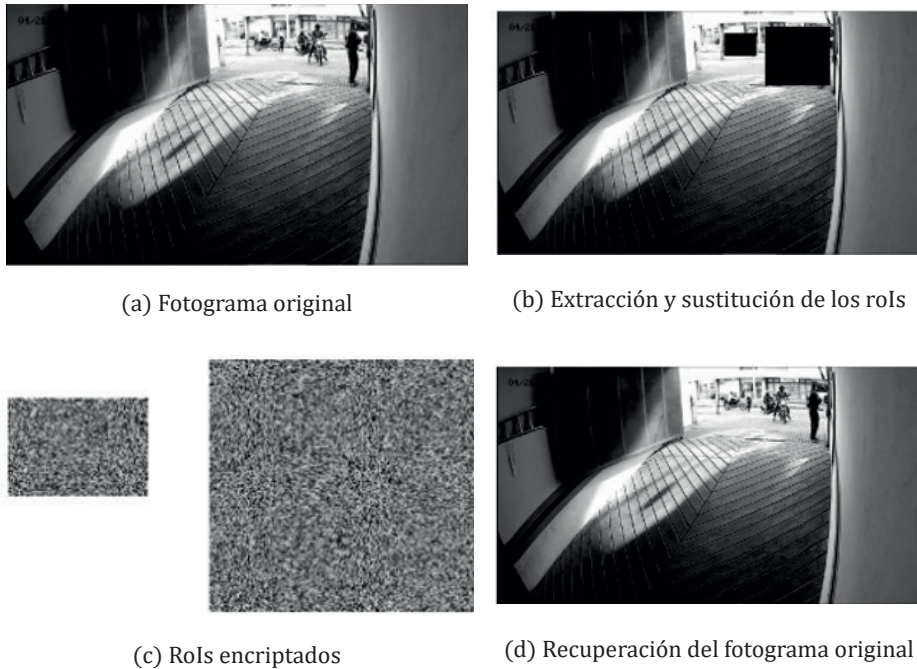
La desencriptación de cada RoI corresponde a la realización del proceso inverso por parte del receptor, a fin de recuperar los RoIs y ubicarlos nuevamente dentro del fotograma. En la figura 40, se utiliza una imagen de la rampa de entrada al parqueadero de la sede de Ingeniería de la Universidad Distrital Francisco José de Caldas para ilustrar el modelo criptográfico propuesto.

Figura 39. Proceso de encriptación de videos de vigilancia con autómatas celulares



Fuente: elaboración propia.

Figura 40. Resultado de la encriptación de videos de vigilancia con autómatas celulares



Fuente: Cortés Martínez *et al.* (2020).

Evaluación del desempeño

El comportamiento de los indicadores de desempeño evaluados para este modelo se perfila como una alternativa de seguridad para ser utilizado en entornos reales.

Para llevar a cabo las pruebas, se tomaron imágenes de las cámaras de video ubicadas en el parqueadero de la sede de Ingeniería de la Universidad Distrital. Con estas se probó la efectividad en la extracción de roIs.

Al analizar el espacio de clave, se evidencia que su tamaño garantiza la resistencia a los ataques de fuerza bruta. También se analizaron los histogramas, en los que se observa una distribución de frecuencias uniforme en las imágenes encriptadas. En cuanto a la entropía, que mide la distribución de valores de gris, se obtienen valores cercanos al óptimo (8) definido en la literatura. A su vez, para verificar la resistencia a un ataque diferencial, se

emplean los indicadores NPCR y UACI. En este caso, los resultados del modelo propuesto estuvieron muy cerca de los valores óptimos planteados en la literatura.

Para medir la correlación de píxeles adyacentes, se tomó una imagen a color y se encontró una correlación casi nula entre los componentes rojo, verde y azul de la imagen cifrada. La sensibilidad de clave se evaluó tomando dos de ellas con un solo bit de diferencia. En este caso, al aplicar UACI y NPCR, se encontró que las imágenes encriptadas con cada una de las claves son completamente diferentes. Los resultados obtenidos evidencian que el modelo presentado es robusto y tiene resistencia a diversos tipos de ataques.

Aplicación de técnicas de esteganografía

Otra de las técnicas de criptografía propuestas en este documento hace referencia a la utilización de un proceso esteganográfico sobre imágenes digitales para ocultar textos cifrados, tomando como base el caos y los autómatas celulares. Esto dio origen a un trabajo de grado para obtener el título de Ingeniero de Sistemas en la Universidad Distrital (Arias Cárdenas, 2020).

El trabajo propuesto consiste en encriptar un mensaje de texto usando el atractor caótico de Tinkerbell y, una vez encriptado, ocultarlo sobre los bordes de una imagen digital. Estos se hallan mediante la utilización de autómatas celulares usando técnicas esteganográficas.

La mezcla de criptografía, esteganografía y autómatas celulares permite asegurar la confidencialidad y la integridad de la información transmitida por los usuarios, frente a un gran número de ataques que se pueden presentar a través del canal de transmisión.

Síntesis del modelo propuesto

Para la parte del cifrado de texto, se obtiene una secuencia pseudoaleatoria a partir de la iteración del atractor caótico de Tinkerbell. Se aplica previamente un desplazamiento tanto en el texto como en la secuencia caótica y, posteriormente, se lleva a cabo la operación XOR.

Por otro lado, se elige la imagen en la que se desea ocultar el texto cifrado y se hallan los bordes de dicha imagen usando autómatas celulares. Sin

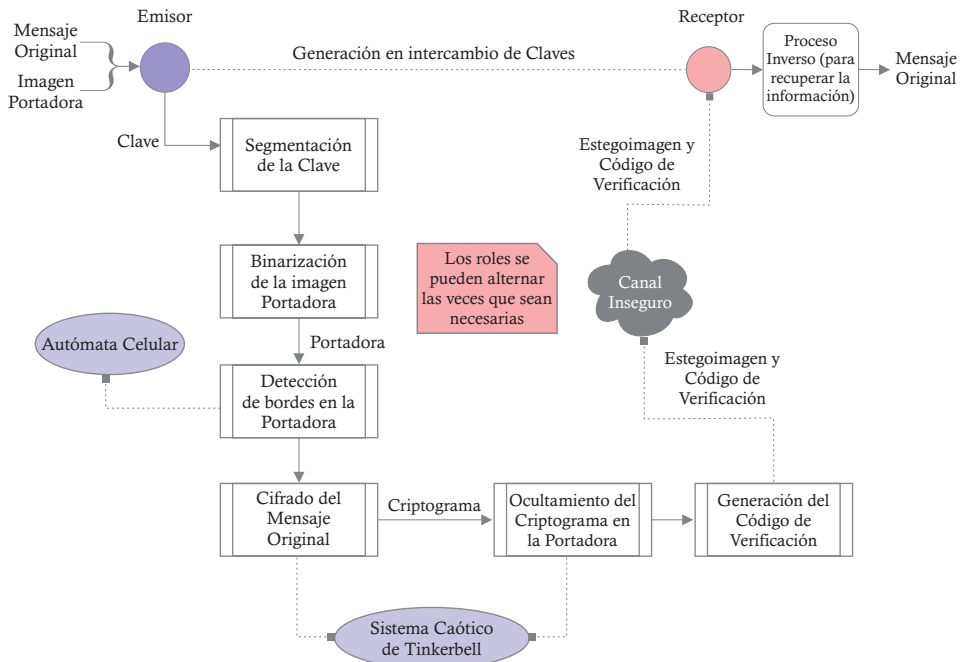
embargo, para ocultar el texto, no se emplean todos los píxeles del borde, sino que son elegidos aleatoriamente usando puntos del espacio de fases del atractor de Tinkerbell, con condiciones iniciales diferentes a las usadas en el proceso de encriptación. Sobre estos píxeles se usa el método esteganográfico de ocultación denominado bit menos significativo o LSB (por sus siglas en inglés) y se aumenta la aleatoriedad mediante el uso del generador congruencial permutado de números pseudoaleatorios (PCG por sus siglas en inglés). Cabe resaltar que todas las condiciones iniciales del atractor de Tinkerbell hacen parte de la clave del criptosistema.

Para el proceso de generación e intercambio de la clave, se utiliza el método propuesto por Diffie y Hellman (1976) y se aplica la función hash SHA3-256 para conseguir un arreglo de 256 bits. Este se segmenta y se emplea para referenciar las condiciones iniciales del atractor de Tinkerbell, así como para determinar el desplazamiento aplicado tanto a la secuencia cifrante como al criptograma y la estegoimagen.

El proceso de recuperación de la información es inverso al anteriormente mencionado. Dado que se guarda la lista de los píxeles donde se ha ocultado la información, se recorren esos mismos píxeles recuperando los bits pertenecientes al texto; posteriormente, se descrypta usando el mismo atractor caótico de Tinkerbell y se obtiene así el texto original. En la figura 41, se presenta una síntesis del modelo propuesto.

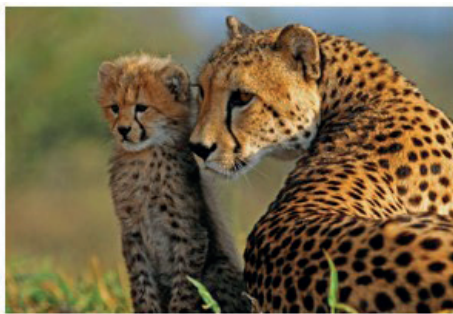
Cabe anotar que la imagen portadora es modificada de manera permanente, es decir, no vuelve a ser la misma imagen original. Sin embargo, los cambios realizados sobre esta para ocultar el texto son casi imperceptibles para el ojo humano, como se evidencia en la figura 42.

Figura 41. Síntesis del proceso de ocultamiento de textos encriptados en una imagen, aplicando técnicas de esteganografía

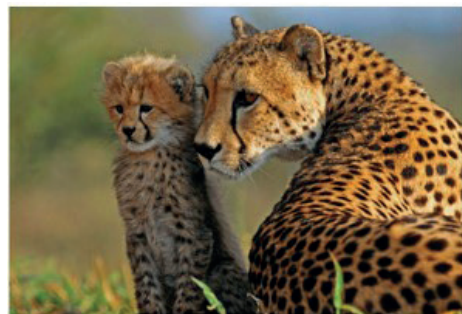


Fuente: Arias Cárdenas (2020).

Figura 42. Ilustración de los resultados del modelo criptoesteganográfico



(a) Imagen original



(b) Estegoimagen

Fuente: Arias Cárdenas (2020).

Evaluación del desempeño

En este caso, la evaluación del desempeño se realiza en dos etapas: la primera sobre el proceso de esteganografía, usando mecanismos que permitan verificar la integridad de la información, y la segunda sobre el sistema en general, donde se evalúa la calidad y la seguridad del sistema propuesto.

Los resultados obtenidos en la evaluación de los procesos de esteganografía y de encriptación fueron cercanos a los óptimos de acuerdo con la literatura. Para evaluar el proceso esteganográfico, se utilizaron parámetros estándar como Proporción Máxima de la Señal de Ruido (PSNR), Error Cuadrático Medio (MSE), Raíz del Error Cuadrático Medio Normalizado (NRMSE), Índice de Similitud Estructural (SSI), Entropía de Shannon, Variación de la Información (IV) e histogramas en los que se evidencia la similitud entre la imagen original y la estegoimagen.

Con el propósito de verificar la robustez del sistema criptográfico, se aplicaron las pruebas de aleatoriedad para cadenas de bits de longitud arbitraria (NIST Test Suite), que comprenden quince indicadores, los cuales determinan el nivel de aleatoriedad de cadenas de bits para evaluar si son susceptibles de ser utilizadas en la definición de algoritmos criptográficos o esteganográficos. En el caso del estudio aquí referenciado, se encontró que casi todas las pruebas fueron superadas. Igualmente, se incluyeron las pruebas de sensibilidad y se determinó que el espacio de clave es amplio e igual a 2^{256} .

En síntesis, los resultados obtenidos con las evaluaciones tanto a nivel criptográfico como esteganográfico se ajustan a los estándares de la literatura reciente. Por tanto, el modelo presentado puede ser considerado como una opción aplicable en situaciones reales.

Aplicación de algoritmos genéticos

Algunos trabajos de la literatura científica dan cuenta de que los algoritmos genéticos abren un camino para consolidar alternativas sobre seguridad de la información. En ese sentido, se planteó un sistema criptográfico

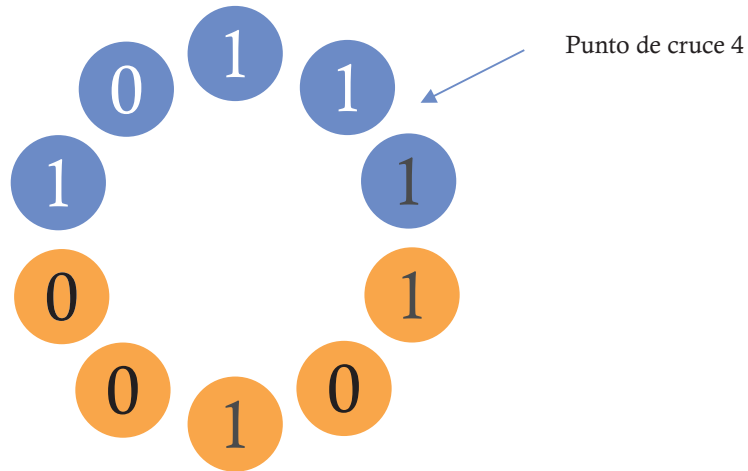
de tipo simétrico enfocado a textos por medio de operadores de selección, cruce y mutación. Esto dio origen a la publicación de un artículo científico (Rodríguez *et al.*, 2019).

Síntesis del modelo propuesto

Dado un texto, se toma la representación de cada caracter en código ASCII, agregando un espacio en blanco entre ellos. Posteriormente, uno a uno, se convierten a binario los dígitos de dicho código. Una vez realizado el proceso anterior, se identifica cada bit como un alelo y cada conjunto de 8 bits como un gen.

Para llevar a cabo el proceso de encriptación, se aplican los operadores de cruce y mutación característicos de los algoritmos genéticos. En este caso, con las cadenas binarias se empleó el cruce de anillo, cada gen padre se cruza con su antecesor y su subsiguiente, de modo que se elimina la población de los genes padre y se reemplaza por sus hijos, como se ilustra en la figura 43.

Figura 43. Cruce tipo anillo



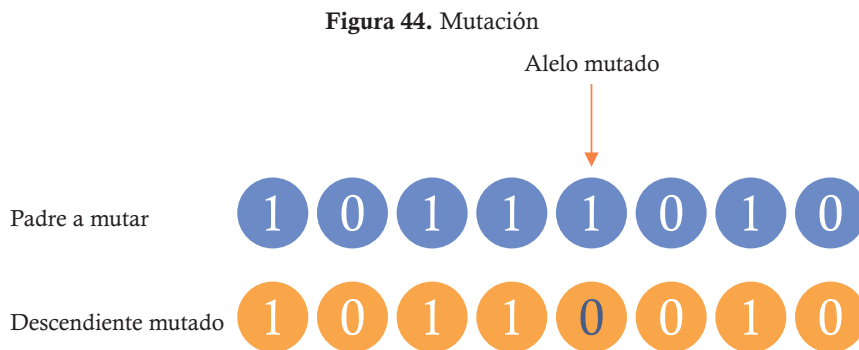
Primer padre: 1011

Segundo padre: 0100

Descendencia con punto de cruce 4: 11010

Fuente: Rodríguez *et al.* (2019).

Para la mutación, se considera la cadena formada por todos los genes descendientes y se toma un número impar n , con el fin de determinar las posiciones a mutar. De esta manera, se obtiene una población que es considerada posible encriptación del mensaje original, sobre la cual se aplica la función *fitness* que evalúa la calidad de la cadena encriptada. En la figura 44, se ilustra este caso.



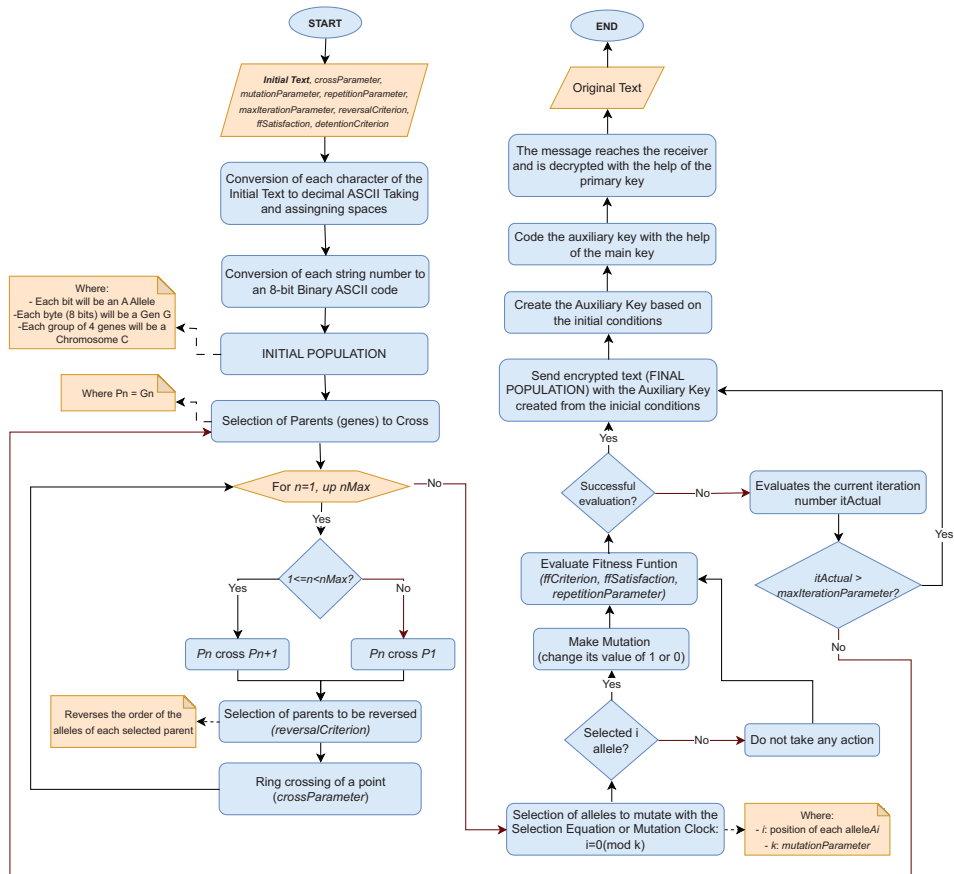
Fuente: Rodríguez *et al.* (2019).

Dependiendo del tamaño del mensaje, se define un criterio para determinar el tamaño de los grupos de alelos. La función *fitness* toma dichos grupos y bajo unos parámetros previamente establecidos, como número de repeticiones máximas permitidas, criterio de detención, cadenas válidas y cantidad de cadenas válidas, determina el nivel de satisfacción de las cadenas generadas, el cual se espera que supere el criterio de detención para ser considerada como la cadena resultante encriptada. En caso contrario, se cambia el tamaño de cada grupo de alelos y se vuelve a aplicar el esquema de cruce, mutación y selección hasta encontrar un nivel de satisfacción mayor o igual al criterio de detención.

Cabe anotar que, aunque el modelo propuesto es de tipo simétrico, se utilizan dos claves. La primera de ellas es una cadena de 16 caracteres en código ASCII. La segunda está formada por 8 parámetros que corresponden a las condiciones iniciales de la función *fitness*, esta es encriptada utilizando

la cadena de los 16 caracteres, con el fin de aumentar el nivel de seguridad del esquema propuesto. Para recuperar el mensaje original, se aplica el mismo proceso en sentido inverso. En la figura 45, se muestra una síntesis del modelo criptográfico.

Figura 45. Síntesis del modelo criptográfico usando algoritmos genéticos



Fuente: Rodríguez *et al.* (2019).

Evaluación del desempeño

Se utilizaron textos de diferentes tamaños para verificar la validez del algoritmo. Modificando los parámetros y criterios propuestos, se calculó la velocidad de ejecución y se llevó a cabo una comparación con los algoritmos DES (Data Encryption Standard), RSA (Rivest, Shamir and Adleman) y AES (Advanced Encryption standard), teniendo presente el tiempo de procesamiento, la escalabilidad y el tamaño de la clave, entre otros. Se concluyó que el algoritmo de este trabajo tiene un mejor rendimiento computacional y que, en esta propuesta, la longitud de la clave es mayor comparada con la de otros métodos analizados.

Referencias

- Al-Mashhadi, H. M. y Abduljaleel, I. Q. (2017). Color Image Encryption Using Chaotic Maps, Triangular Scrambling, with DNA Sequences. *2017 International Conference on Current Research in Computer Science and Information Technology (ICCRIT)*, 93-98. <https://doi.org/10.1109/CRCSIT.2017.7965540>
- Arias Cárdenas, M. (2020). *Criptosistema esteganográfico a través de imágenes digitales para el cifrado de texto usando teoría del caos y autómatas celulares* [Tesis de pregrado, Universidad Distrital Francisco José de Caldas]. <https://repository.udistrital.edu.co/handle/11349/27985>
- Bakhshandeh, A. y Eslami, Z. (2013). An Authenticated Image Encryption Scheme Based on Chaotic Maps and Memory Cellular Automata. *Optics and Lasers in Engineering*, 51(6), 665-673. <https://doi.org/10.1016/j.optlaseng.2013.01.001>
- Barragán Guzmán, P. A. y Arias Sarria, C. A. (2019). *Cifrado de videos transmitidos en tiempo real utilizando el atractor de Lorenz* [Tesis de pregrado, Universidad Distrital Francisco José de Caldas]. [https:// repository.udistrital.edu.co/handle/11349/16325](https://repository.udistrital.edu.co/handle/11349/16325)
- Boudali, A., Said, N. H. y Ali-Pacha, A. (2022). A New Symmetrical Cryptosystem Based Cellular Automata and Chaotic Map Function. *Journal of Discrete Mathematical Sciences and Cryptography*, 25(5), 1435-1455. <https://doi.org/10.1080/09720529.2021.2020427>

- Cortés Martínez, L. M., Alvarado Nieto, L. D. y Amaya Barrera, E. I. (2020). Composite Cellular Automata Based Encryption Method Applied to Surveillance Videos. *DYNA*, 87(213), 212-221. <https://doi.org/10.15446/dyna.v87n213.81859>
- Diffie, W. y Hellman, M. (1976). New Directions in Cryptography. *IEEE Transactions on Information Theory*, 22(6), 644-654. <https://doi.org/10.1109/TIT.1976.1055638>
- ElSharkawy, M. I. (2017). Integrating and Securing Video, Audio and Text Using Quaternion Fourier Transform. *International Journal of Communication Networks and Information Security (IJCNIS)*, 9(3). <https://doi.org/10.17762/ijcnis.v9i3.2634>
- Jose, J. y Chowdhury, D. R. (2017). Investigating Four Neighbourhood Cellular Automata as Better Cryptographic Primitives. *Journal of Discrete Mathematical Sciences and Cryptography*, 20(8), 1675-1695. <https://doi.org/10.1080/09720529.2016.1160533>
- Kumar, A. y Kumar Sharma, S. (2022). Information Cryptography Using Cellular Automata and Digital Image Processing. *Journal of Discrete Mathematical Sciences and Cryptography*, 25(4), 1105-1111. <https://doi.org/10.1080/09720529.2022.2072437>
- Mendez, L., Ladino, D., Amaya, I. y Alvarado, D. (2023). A new multimedia cryptosystem using chaos, quaternion theory and modular arithmetic. *Multimedia Tools and Applications*, (82), 35149-35181. <https://doi.org/10.1007/s11042-023-14475-1>
- Méndez López, L. E. y Ladino Torres, D. E. (2019). *Sistema para encriptación de archivos multimedia a partir de atractores caóticos y teoría de cuaterniones* [Tesis de pregrado, Universidad Distrital Francisco José de Caldas]. <https://repository.udistrital.edu.co/handle/11349/15848>
- Nagaraj, S., Raju, G. S. V. P. y Koteswara Rao, K. (2015). Image Encryption Using Elliptic Curve Cryptography and Matrix. *Procedia Computer Science*, 48, 276-281. <https://doi.org/10.1016/j.procs.2015.04.182>
- Pecora, L. M. y Carroll, T. L. (1990). Synchronization in Chaotic Systems. *Physical Review Letters*, 64(8), 821-824. <https://doi.org/10.1103/PhysRevLett.64.821>
- Rodríguez, J., Corredor, B. y Suárez, C. (2019). Genetic Operators Applied to Symmetric Cryptography. *International Journal of Interactive Multimedia and Artificial Intelligence*, 5(7), 39-49. <https://doi.org/10.9781/ijimai.2019.07.006>

- Santos Bustos, D., Amaya Barrera, I. y Suárez Parra, C. A. (2020). Algoritmo de encriptación para imágenes a color basado en sistemas caóticos. *Ingeniería*, 25(2), 144-161. <https://doi.org/10.14483/23448393.15530>
- Valluri, M. R. y Narayan, S. V. (2016). Quaternion Public Key Cryptosystems. *2016 World Congress on Industrial Control Systems Security (WCICSS)*, London, UK. <https://doi.org/10.1109/WCICSS.2016.7882612>
- Wu, X., Zhu, B., Hu, Y. y Ran, Y. (2017). A Novel Color Image Encryption Scheme Using Rectangular Transform-Enhanced Chaotic Tent Maps. *IEEE Access*, 5, 6429-6436. <https://doi.org/10.1109/ACCESS.2017.2692043>

Sobre los autores

Luz Deicy Alvarado-Nieto

Profesora titular en la Facultad de Ingeniería de la Universidad Distrital Francisco José de Caldas desde 1995. Ingeniera de Sistemas de la misma universidad, Magíster en Ingeniería de Sistemas de la Universidad Nacional de Colombia, Doctora en Informática con énfasis en Ciencias de la Computación e Inteligencia Artificial de la Universidad de Oviedo, España. Actualmente dirige el Grupo de Complejidad de la Universidad Distrital Francisco José de Caldas (ComplexUD).

Correo electrónico: lalvarado@udistrital.edu.co

ORCID: <https://orcid.org/0000-0002-1305-3123>

Isabel Amaya-Barrera

Profesora titular en la Facultad de Ingeniería de la Universidad Distrital Francisco José de Caldas desde 2001. Licenciada en Matemáticas de la misma universidad, Colombia, Especialista en Matemática Aplicada de la Universidad Sergio Arboleda, Magíster en Ciencias Matemáticas de la Universidad Nacional de Colombia. Es miembro del Grupo de Complejidad de la Universidad Distrital Francisco José de Caldas (ComplexUD).

Correo electrónico: iamaya@udistrital.edu.co

ORCID: <https://orcid.org/0000-0002-8845-5901>

César Augusto Suárez Parra

Profesor asistente en la Facultad de Ingeniería de la Universidad Distrital Francisco José de Caldas desde 1999. Ingeniero Mecánico de la Universidad INCCA de Bogotá, Colombia, Magíster en Materiales y Procesos de Manufactura de la Universidad Nacional de Colombia. Es miembro del Grupo de Complejidad de la Universidad Distrital Francisco José de Caldas (ComplexUD).

Correo electrónico: casuarezp@udistrital.edu.co

ORCID: <https://orcid.org/0000-0002-2861-7788>

EC



Este libro se terminó de imprimir en septiembre de 2024
en XPRESS ESTUDIO GRÁFICO Y DIGITAL S. A. S.,
Bogotá, Colombia.