ISSN 1870-4069

Método criptográfico simétrico utilizando teoría del caos, operaciones sobre ADN y raíces de funciones no lineales

Luis René Marcial Castillo, Erika Leonor Basurto Munguia,
Marcela Rivera Martínez, María de Lourdes Sandoval Solis

Benemérita Universidad Autónoma de Puebla, Facultad de Ciencias de la Computación, Puebla, México

{luis.marcial, marcela.rivera, maria.sandoval}@correo.buap.mx, iamdleonor@gmail.com

Resumen. En este trabajo se propone un algoritmo para criptografía simétrica el cual se basa en funciones caóticas, enmascaramiento basado en adición y sustracción sobre el ácido desoxirribonucleico y raíces de funciones no lineales. El algoritmo genera dos llaves; la primera es generada por la función caótica y la segunda por la función no lineal. La implementación computacional del algoritmo realizada en Octave permite cifrar y descifrar texto de cualquier dimensión con un alto nivel de seguridad apoyado esto en un análisis de sensibilidad que muestra la resistencia ante ataques exhaustivos.

Palabras clave: Criptografía simétrica, funciones caóticas, secuencias de ADN, funciones no lineales.

Symmetric Cryptographic Method Using Chaos Theory, Operations on DNA and Roots of Nonlinear Functions

Abstract. In this work we propose an algorithm for symmetric cryptography based on chaotic functions, masking based on addition and subtraction on deoxyribonucleic acid and roots of nonlinear functions. The algorithm generates two keys; the first one is generated by the chaotic function and the second by the nonlinear function. The computational implementation of the algorithm realized in Octave allows to encrypt and decryption text of any dimension with a high level of security supported this in a sensitivity analysis that shows the resistance to exhaustive attacks.

Keywords: Symmetric cryptography, chaotic functions, DNA sequences, nonlinear functions.

1. Introducción

La confidencialidad de las comunicaciones es de suma importancia en la sociedad moderna. La industria, gobierno y particulares confían en que la tecnología les garantice

que el intercambio de datos sea seguro de modo que no se permita a terceros acceder al contenido de tal comunicación, la cuestión de la confidencialidad ha sido dejada al campo de la criptografía [4]. Goldreich transcribe el planteamiento de la criptografía como "el problema de proveer comunicación secreta sobre medios inseguros" [8].

La criptografía es una herramienta muy útil cuando se desea tener seguridad informática, es decir, cuando se cuenta con un medio para garantizar las propiedades de confidencialidad, para lograrlo, se crean mecanismos que garanticen en cierta medida las propiedades de disponibilidad, integridad y confidencialidad. La disponibilidad se refiere a que la información siempre este presente, la integridad significa no perder información, la confidencialidad se puede lograr usando mecanismos que aunque sea robada la información, permita el no acceso a esta o garantice de alguna forma no poder llegar a ella, hasta que pierda su valor. Estos mecanismos permiten ver si la información ya creada ha sufrido o no alguna modificación no autorizada. El criptoanálisis, también llamado "criptología" es la disciplina contraria a la criptografía, se encarga de analizar la información cifrada para revelar la información original sin necesidad de las claves secretas y de esta forma romper los procedimientos previamente establecidos por la criptografía, el criptoanálisis lo usan los investigadores como una forma de probar las fortalezas o debilidades de los cripto sistemas [6].

Desde la década de los 90's muchos investigadores han notado que existe una importante relación entre el caos y la criptografía: muchas propiedades de los sistemas caóticos tienen sus correspondientes contrapartes en los cripto sistemas tradicionales. Los sistemas caóticos pueden conocer sus ecuaciones y sus condiciones iniciales fijas, sin embargo, la más mínima variación provoca una evolución radical en su comportamiento [4]. La teoría de funciones y mapas caóticos se presentan en varios trabajos. Fuan, Mengb, Zhanb, Zhuc, Laud, Tsed y Mae en el 2013 [5] proponen un esquema de protección de imágenes médicas basado en mapas caóticos; Gao y Chen en el 2008 [7] proponen un algoritmo nuevo de permutación de pixeles; Huang y Nien en el 2009 [10] proponen un sistema multi-caótico basado en el mismo principio de permutación de Gao y Chen; Patidar, Pareek y Sud en el 2009 [15] proponen un cifrado tipo substitución-difusión basado en mapas logísticos y caóticos; Rhouma, Meherzi y Belghith en el 2009 [16] proponen el cifrado de imágenes a color basado en mapas caóticos; Sun, Liu y Li en el 2008 [17] proponen un esquema de encriptación basado en mapas caóticos espaciales; Tong y Cui en el 2009 [19] proponen un generador de cifrado de secuencias caóticas con componentes dinámicos; Wong, Kwok y Law en el 2008 [20] proponen un esquema de encriptación basado en el mapa caótico estándar; Xiao y Xia en el 2009 [21] proponen un esquema de encriptación usando mapas de permutación; Xu, Wang y Yang en el 2008 [22] proponen una mejora en el algoritmo de encriptación de imágenes que usan mapas caóticos y Ye en el 2009 [23] propone un cripto sistema basado en las matrices de Töplitz y Hankel.

En lo que se refiere a la teoría de secuencias de ácido desoxirribonucleico (ADN) se presenta en varios trabajos. Teree, Vaida, Alboaie y Chiorean, en el 2011 [18] proponen el uso de ADN para criptografía simétrica; Anwarl, Paul y Singh en el 2014 [2] hacen una revisión de la forma en que se ha realizado la transmisión de mensajes basada en el uso de ADN; Javheri y Kulkarni en el 2014 [13] proponen también un algoritmo criptográfico para la comunicación segura de datos basado en ADN; Anil y

Chirakkarottu en el 2014 [1] presentan un método de encriptación para el iris del ojo humano basado también en operaciones de ADN.

Como lo muestran los trabajos mencionados, las funciones caóticas y las operaciones de ADN han llamado la atención de varios investigadores y las han utilizado para diseñar sus algoritmos criptográficos.

La propuesta de este trabajo además de utilizar resultados sobre funciones caóticas para generar la primera llave como lo muestran los trabajos mencionados y operaciones de ADN para realizar el enmascaramiento, agrega el uso de raíces de una función no lineal, con lo cual se refuerza la seguridad, al proporcionarle al algoritmo una segunda llave, con un costo muy bajo.

En la siguiente sección se presentan los aspectos matemáticos usados en el algoritmo, en la sección 3 se muestra y detalla el algoritmo propuesto, la sección 4 proporciona las pruebas del cripto sistema. La sección número 5 presenta las conclusiones y finalmente se listan las referencias utilizadas en el desarrollo de este trabajo.

2. Aspectos matemáticos

2.1. Teoría del caos

Puede decirse que la dinámica caótica inició con el trabajo del matemático Francés Henri Poincaré a finales del siglo XIX. La motivación de Poincaré fue promovida por el problema de las orbitas de tres cuerpos celestes experimentando atracción gravitacional mutua (por ejemplo, una estrella y dos planetas). Poincaré fué capaz de mostrar que orbitas muy complicadas eran posibles (ahora llamadas caóticas). No obstante, la posibilidad de caos en sistemas físicos reales no fue ampliamente apreciada sino hasta la actualidad, mucho del crédito por este cambio es atribuido a la extensa solución numérica de sistemas dinámicos en computadoras digitales [14].

En los últimos años, los hilos del caos y la dinámica no lineal se han esparcido a través de disciplinas científicas como una intrincada red araña. Caos y dinámica no lineal han provisto de nuevas herramientas teóricas y conceptuales que permiten capturar, entender y enlazar los comportamientos complejos de sistemas simples (el tipo de comportamiento llamado caos en la ciencia contemporánea) [9]. Se puede decir, que una función caótica es una función matemática que describe un sistema dinámico no lineal complejo cuya evolución en el tiempo hace imposible la predicción a largo plazo, luciendo errático y casi aleatorio. Los ejemplos más notorios de las características del caos son el llamado efecto mariposa y la impredecibilidad de órbitas pseudo-aleatorias, generadas por ecuaciones deterministas. Estos fenómenos, así como otros relacionados con el caos, han sido tradicionalmente asociados a mecanismos de confusión y difusión los cuales son la base principal de un buen sistema criptográfico.

El elemento central de todos los sistemas caóticos es el concepto de iteración. El estado actual del sistema es una función determinística del estado o valor anterior. Formalmente, una correspondencia caótica se especifica por medio de la expresión:

$$x_{k+1} = f(x_k). \tag{1}$$

La fórmula (1) muestra la expresión de la correspondencia caótica, donde *f* es una función no lineal. En general, la teoría de caos surge de la necesidad de modelar mecanismos físicos tales como la predicción del tiempo atmosférico, la evolución de la población, la dinámica de fluidos, la teoría de gases, predicción del tiempo, etc.

Un sistema caótico es un sistema dinámico, no lineal, determinístico que muestra una dependencia muy sensible a las condiciones iniciales y presenta una evolución a través de un espacio de fase que parece ser aleatorio [8]. Estas propiedades proporcionan un potencial para aplicaciones en criptografía ya que las predicciones a largo plazo de los sistemas caóticos son muy difíciles [11].

Existen varias funciones caóticas usadas en el campo de la criptografía, pero de las analizadas por Teree, Vaida, Alboaie y Chiorea, 2011 [18] la *cross chaotic map* es la que les dio mejores resultados, razón por la cual es usada en este trabajo para generar la primera llave. Tal función es definida por las ecuaciones (2) y (3):

$$x_{i+1} = 1 - \eta y_i^2 \text{ , donde } \eta \text{ es una constante,} \quad (2)$$

$$y_{i+1} = \cos\left(k \cos^{-1}\left(x_i\right)\right), \text{ donde } k \text{ es una constante.} \quad (3)$$

2.2. Secuencias de ADN

El ácido desoxirribonucleico es el material genético de todos los organismos celulares y de casi todos los virus. El ADN lleva la información necesaria para dirigir la síntesis de proteínas y la replicación. Cada molécula de ADN está constituida por dos cadenas o bandas formadas por un elevado número de compuestos químicos llamados nucleótidos. Estas cadenas forman una especie de escalera retorcida que se llama doble hélice. Cada nucleótido está formado por tres unidades: una molécula de azúcar llamada desoxirribosa, un grupo fosfato y uno de cuatro posibles compuestos nitrogenados llamados bases, las cuales son: Adenina (A), Guanina (G), Tiamina (T) y Citosina (C). Conforme al complemento presentado por Watson y Crick, la Adenina es complementada por Tiamina, y Guanina con Citosina [1].

Tabla 1. Regla de adición para el ADN.

+	A	C	G	T
A	A	C	G	T
C	C	G	T	A
G	G	T	A	C
T	T	A	C	G

En este método la Adenina se codifica con 00, la Tiamina con 11, la Guanina con 10 y la Citosina con 01 [18]. La regla de adición (suma binaria) para el ADN es realizada como lo muestra la tabla 1, por ejemplo, si se desea realizar la operación T + T el resultado sería G, ya que la suma binaria 11 + 11 es igual a 110 pero al desechar el bit más significativo (más a la izquierda) se obtiene el número binario 10 que corresponde a G. La regla de sustracción (resta binaria en complemento a 2) es mostrada en la tabla 2, por ejemplo, el resultado de la operación T – T es A, ya que el complemento a 2 de

T es: 01 y al sumar T + 01 resulta que 11 + 01 = 100, en donde se desecha el bit más significativo y el resultado es 00 que corresponde a la letra A.

Tabla 2. Regla de sustracción para el ADN.

-	A	C	G	T
A	A	T	G	C
C	C	A	T	G
G	G	C	A	T
T	T	G	C	A

Las operaciones de suma y sustracción de ADN son usadas en este trabajo.

2.3. Raíces en funciones no lineales

Cuando se desea encontrar algún x que resuelva la ecuación no lineal $f(x) = 0$, se puede usar algún método iterativo. Uno de los métodos más conocidos y eficientes es el método de Newton. El método de Newton se obtiene de la forma siguiente [3]:

Paso 1. Se realiza la expansión de Taylor de grado 1 a la función $f(x)$ alrededor de un punto local x_k y se iguala a cero, es decir,

$$f(x) = f(x_k) + f'(x_k)(x - x_k) = 0. \tag{4}$$

Paso 2. Se despeja a la variable x de la ecuación (4) y se obtiene:

$$x = x_k - \frac{f(x_k)}{f'(x_k)}. \tag{5}$$

Paso 3. El siguiente punto x_{k+1} se obtiene simplemente al sustituir x por x_{k+1} en la ecuación 5, obteniendo:

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}. \tag{6}$$

Una de las ventajas del método de Newton es que si r es una raíz simple de la función no lineal $f(x) = 0$, entonces, converge cuadráticamente, en caso contrario converge linealmente. Es importante que la aproximación inicial este cerca para que se cumplan las ventajas mencionadas. Una desventaja que tiene el método es que si no se conoce analíticamente la derivada de la función $f(x)$ debe aproximarse de forma numérica, lo cual no pasa para la función $f(x)$ que se usa en este trabajo.

Algoritmo 1. Algoritmo del método de Newton para encontrar alguna de las raíces de la función no lineal $f(x)$.

Inicio

Dado una aproximación inicial x_i a la raíz de la función no lineal $f(x)$.

$$x_n = x_i - \frac{f(x_i)}{f'(x_i)}.$$

Mientras no se cumpla la condición de paro

$$x_i = x_n$$

$$x_n = x_i - \frac{f(x_i)}{f'(x_i)}.$$

Fin del ciclo Mientras

Fin

Para el algoritmo criptográfico propuesto se encuentra la raíz de la función no lineal $f(x) - valorascii$, donde *valorascii* es el código ASCII asociado al carácter que se desea cifrar.

3. Algoritmo propuesto

El algoritmo que se propone se basa en las funciones caóticas, operaciones sobre ADN y raíces de funciones no lineales. El cripto sistema consta de los algoritmos de cifrado y descifrado sobre texto. A continuación se presentan tales algoritmos.

Algoritmo 2. Algoritmo de cifrado sobre texto.

Inicio

1. Se ingresa el texto a cifrar T, y se encuentra su longitud, dejando el resultado en n.
2. Se calcula el valor de m como: $m = \lceil \sqrt{n} \rceil$, donde $\lceil x \rceil$ denota el menor entero que es mayor o igual al valor real x.
3. En caso de ser necesario, se agrega basura al final del arreglo T de modo que se obtenga un nuevo arreglo T_e cuya longitud es m x m.
4. Se convierten a decimal los caracteres del arreglo T_e usando para ello los códigos ASCII de tales caracteres, obteniendo el nuevo arreglo T_d.
5. El arreglo T_d se apila en las columnas de una matriz I cuya dimensión será m x m.
6. Codificar los elementos de I a binario usando 8 bits para representar cada dígito, dejando el resultado en la matriz I_b.
7. Se codifica la imagen binaria I_b a una secuencia de ADN y el resultado se deja en una matriz denotada con L.
8. Se construye la primera llave de la forma siguiente:
   Se calculan los valores k_1 y k_2 como:

$$k_1 = \frac{1}{126} \bmod \left(\sum_{i=1}^{m/2} \sum_{j=1}^{m} I_{i,j}, 256 \right),$$

$$k_2 = \frac{1}{126} \bmod \left(\sum_{i=m/2+1}^{m} \sum_{j=1}^{m} I_{i,j}, 256 \right).$$

   Se generan de forma aleatoria los valores: x_0, y_0 sobre el intervalo real [0,1].
   Se calcula el valor inicial x_0 de la secuencia caótica como: (se repite el mismo proceso pero usando y_0 en lugar de x_0) [18]

$$x_0 = x_1 + k_1$$

   Si $x_0 > 1$ entonces $x_0 = \bmod(x_0, 1)$.

Se genera $X = (x_1,\ldots,x_m)$ como vector columna y $Y = (Y_1,\ldots,Y_{8m})$ como vector fila usando la formula caótica dada por las ecuaciones (2) y (3) con $\eta = 2$ y $k = 6$.

Se multiplica el vector columna X por el vector fila Y, obteniendo una matriz M de dimensión $m \times 8m$.

Se convierte la matriz M a binario usando:

$$M_b(i, j) = \begin{cases} 0, & \text{Si } M(i, j) < 0 \\ 1, & \text{Si } M(i, j) \geq 0. \end{cases}$$

Se codifica M_b a cadenas de ADN obteniendo la matriz K.

9. Se aplica la adición de ADN como lo dicta la tabla 1.
   $NADN = L + K$.
10. La matriz obtenida $NADN$ se pasa a binario obteniendo N_b.
11. Se convierte N_b a valores enteros agrupando cada 8 bits para formar un valor entero entre 0 y 255, obteniendo una matriz C_p de dimensión $m \times m$.
12. Generar de modo aleatorio en el rango [0, 1] los valores de las constantes que componen a la función no lineal $f(x)$ seleccionada, estos valores constantes son la segunda llave del proceso de cifrado.
13. Para cada valor ASCII de C_p, se aplica el método de Newton para encontrar alguna c que sea raíz de la función no lineal $f(x)$ — *valor ASCII*.
14. Pasar todos los valores reales c encontrados por el método de Newton a una cadena hexadecimal c_b que se compone de 64 bits si se usan reales dobles, o de 32 bits si se usan reales simples, dejando las cadenas hexadecimales en el arreglo C.

Fin

El cifrado final se encuentra en C y es una cadena en formato hexadecimal.

Algoritmo 3. Algoritmo de descifrado sobre texto.

Inicio

1. Se pasan las cadenas hexadecimales almacenadas en C a números decimales, dejando el resultado en el arreglo D_d.
2. Los valores de D_d se evalúan en la función no lineal $f(x)$ que contiene constantes que forman la segunda llave, las evaluaciones se dejan en D_{de}.
3. Se redondean al entero más cercano los valores de D_{de} dejando el resultado en D_{de}.
4. Se pasa la matriz D_{de} a valores binarios, obteniendo D_b.
5. Se codifica D_b a valores de ADN obteniendo $DADN$.
6. Se aplica la regla de sustracción de ADN dada por la tabla 2.
   $L_2 = K - DADN$, donde K es la primera llave generada en el proceso de cifrado.
7. Se convierte L_2 a binario obteniendo L_{2b}.
8. Se convierte L_{2b} a decimal, obteniendo DI_d.
9. Se pasa DI_d a un vector de longitud $m \times m$ desapilando por columnas, dejando el resultado en D_e.
10. Se elimina la basura tomando solo los primeros n elementos de D_e, dejando el resultado en DT_d.
11. Se convierte a caracter cada uno de los valores de DT_d dejando el resultado en T.

Fin

Al finalizar el algoritmo de descifrado se obtiene el texto original en T.

4. Pruebas

El algoritmo criptográfico simétrico propuesto se desarrolló bajo el lenguaje de programación OCTAVE [24] y las pruebas se realizaron en una computadora hp con 16 Gb de memoria RAM y un procesador intel core i7 a 2.6 GHz.

4.1. Cifrado y descifrado de texto

En la prueba que se presenta a continuación se usa como función no lineal a:

$$f(x) = ax^3 + b\cos(x), \qquad (7)$$

donde las constantes reales *a* y *b* se generan de forma aleatoria sobre [0,1]. El algoritmo 2, obtiene los siguientes resultados cuando cifra el texto: "*The 10th International Congress on Intelligent and Information Technologies 2016".*

El paso 1, obtiene $n = 80$ y el paso 2, obtiene $m = 9$.

El paso 3, agrega al final del texto un carácter % como basura. T_e = The 10th International Congress on Intelligent and Information Technologies 2016%

El paso 4, obtiene (al pasar a decimal los caracteres de T_e) T_d = 84 104 101 32 49 48 116 104 32 73 110 116 101 114 110 97 116 105 111 110 97 108 32 67 111 110 103 114 101 115 115 32 111 110 32 73 110 116 101 108 108 105 103 101 110 116 32 97 110 100 32 73 110 102 111 114 109 97 116 105 111 110 32 84 101 99 104 110 111 108 111 103 105 101 115 32 50 48 49 54 37.

El paso 5, apila los datos de T_d en una matriz *I* de 9 x 9.

$$I = \begin{matrix} 84 & 73 & 111 & 114 & 110 & 116 & 111 & 84 & 105 \\ 104 & 110 & 110 & 101 & 116 & 32 & 114 & 101 & 101 \\ 101 & 116 & 97 & 115 & 101 & 97 & 109 & 99 & 115 \\ 32 & 101 & 108 & 115 & 108 & 110 & 97 & 104 & 32 \\ 49 & 114 & 32 & 32 & 108 & 100 & 116 & 110 & 50 \\ 48 & 110 & 67 & 111 & 105 & 32 & 105 & 111 & 48 \\ 116 & 97 & 111 & 110 & 103 & 73 & 111 & 108 & 49 \\ 104 & 116 & 110 & 32 & 101 & 110 & 110 & 111 & 54 \\ 32 & 105 & 103 & 73 & 110 & 102 & 32 & 103 & 37. \end{matrix}$$

El paso 6, pasa a binario los datos de *I* usando 8 bits para cada dato, y el paso 7 codifica las cadenas binarias a secuencias de ADN.

L = CCCACAGCCGTTCTAGCGTGCTCACGTTCCCACGGC
CGGACGTGCGTGCGCCCTCAAGAACTAGCGCCCGCC
CGCCCTCACGACCTATCGCCCGACCGTCCGATCTAT
AGAACGCCCGTACTATCGTACGTGCGACCGGAAGAA
ATACCTAGAGAAAGAACGTACGCACTCACGTGATAG
ATAACGTGCAATCGTTCGGCAGAACGGCCGTTATAA
CTCACGACCGTTCGTGCGCTCAGCCGTTCGTAATAC
CGGACTCACGTGAGAACGCCCGTGCGTGCGTTATCG
AGAACGGCCGCTCAGCCGTGCGCGAGAACGCTAGCC.

El paso 8, genera la primera llave usando la función caótica. La matriz final en cadenas de ADN es:

K = GAGCCAGGTCCAGATTATTGTTTATAGTTGGCAACC
CTCGGTCCAGGTCTAATAACAAATATCAACCGTTGG
CTCGGTCCAGGTCTAATAACAAATATCAACCGTTGG
GAGCCAGGTCCAGATTATTGTTTATAGTTGGCAACC
CTCGGTCCAGGTCTAATAACAAATATCAACCGTTGG
CTCGGTCCAGGTCTAATAACAAATATCAACCGTTGG
GAGCCAGGTCCAGATTATTGTTTATAGTTGGCAACC
GAGCCAGGTCCAGATTATTGTTTATAGTTGGCAACC
GAGCCAGGTCCAGATTATTGTTTATAGTTGGCAACC.

El paso 9, aplica la adición de ADN como lo indica la tabla 1.

NADN = TCTCGAATATATTTTCCCGAAGAAAGCGATTCCGTG
GCTGTCATCACCGCCCATCCAGATCGCGCTGTACTT
GCGTTGGCCAGAGGATAGCGCGAACCACCTCCAGGC
GGGCGGTTATAATTTGCCGGACGGAGGAAAACAGCC
CGCTTGCTAAGTCCAAAGTCCGCTCGGACTAATGGA
CGCGTCATCGGGGCTTAGGGAGATCCTCCTACTGGG
TTTCGGGTATATTGGCCCACATCCAGCGAACCATCG
TGACGTTGATAGGGTTCCATACGGAGCCAACAATGT
GGGCGGATATGTTACACCGAACAGTGGTAATAAGGG.

El paso 10, pasa a binario el resultado *NADN* y el paso 11 pasa cada 8 bits a un número decimal.

C_p = 221 131 51 253 88 32 38 61 110
158 211 69 149 53 35 102 123 31
155 233 72 163 38 96 81 117 41
169 175 48 254 90 26 40 1 37
103 231 11 80 45 103 104 112 232
102 211 106 159 42 35 93 113 234
253 171 51 233 81 53 38 5 54
225 190 50 175 83 26 37 4 59
169 163 59 196 88 18 235 12 42.

El paso 12, genera los valores *a*, *b* de la segunda llave:

a = 7.638979442864783e-001, *b* = 7.593273831310963e-001.

El paso 13, encuentra las raíces de la función no lineal usando el método de Newton, y el paso 14 escribe los valores del paso 13 de doble precisión a formato hexadecimal.

C =
401a6d303b77b38f 40179ecda782ef8e 40177841d2962af6 40182836cc1ed798
40147dbfaaf0c981 40146cdd2d199c2b 401ba7149539747c 401a95febcf0d786
40182836cc1ed698 401630f5884fd08a 401a0501c5ea7f4c 401ae6335d64bd7a
401870bb03277bd2 401ad2523455744b 401a0501c5ea7f4d 40184091bcaeacd3

40191f36f44d7ef5 4017de01cf6a6d59 40104672751f056e 4011f633ff5abedb
401236bb5cb5ef59 400fea5291610c10 4003cf1eeaee4b97 4014afc6f0595b09
40104672751f0acc 40102bb58d026495 401110e9c8ccec37 401ba7149539747e
401729b0e23f2caa 4017de01cf6c649d 401bb074bef5f47e 4012dab1fec3491e
4017ab8bf2451582 401ae6335d65643a 401870bb03277e89 4019627e794f1955
4013745c2b909c74 40107ae8bc0c3cc9 400d966499343aa9 4013995a5c45abf0
400f40ed7d8632bc 400e8feb2ed2e8c9 4012ee6e7f0fee9d 4013156fb5e5638a
4013745c2b909c73 400bfd084b717326 400ccf8fa5828904 4014053e2faf2c66
400a2c850e84dd64 40147dbfaaf0cbb2 400ccf8fa5873228 40107ae8bc0c35fa
400a2c850ea76eec 40073e9bc8110182 400d966499343877 40146cdd2d197dfc
4012ee6e7f0fdb6d 400e1538b7c2c99d 40148e872d073835 4013cfdc1a3f7818
400d96649934387e 400d55503bd38da2 401af9f7de35e7df 401140b086812c78
4015bb4c74527537 40155fc5b998242f 3fec0a9df4a76336 40151128dfa70a65
4015210f82ab020b 3ffe6999dfed1963 3ffc1ae17f876d80 4004614646a369c2
4014f114bd52a05d 400bb3ef591b1c04 400e53109ebf5cef 400d55503bbf82e5
401adc46643bdabc 401af0192c872d9b 401094a79cf91721 401110e9c8cceb79
400e8feb2ed2e9ed.

C contiene el mensaje cifrado en formato hexadecimal.

El algoritmo 3 de descifrado sobre texto, obtiene los siguientes resultados.

El paso 1, pasa los valores hexadecimales de *C* a decimales, el paso 2 evalúa los decimales usando la ecuación 7 y el paso 3 redondea los valores del paso 2 al entero más cercano obteniendo:

D_{de} =

221 131 51 253 88 32 38 61 110
158 211 69 149 53 35 102 123 31
155 233 72 163 38 96 81 117 41
169 175 48 254 90 26 40 1 37
103 231 11 80 45 103 104 112 232
102 211 106 159 42 35 93 113 234
253 171 51 233 81 53 38 5 54
225 190 50 175 83 26 37 4 59
169 163 59 196 88 18 235 12 42.

El paso 4, pasa D_{de} a binario y el paso 5 pasa a cadenas de ADN.

DADN = TCTCGAATATATTTTCCCGAAGAAAGCGATTCCGTG
GCTGTCATCACCGCCCATCCAGATCGCGCTGTACTT
GCGTTGGCCAGAGGATAGCGCGAACCACCTCCAGGC
GGGCGGTTATAATTTGCCGGACGGAGGAAAACAGCC
CGCTTGCTAAGTCCAAAGTCCGCTCGGACTAATGGA
CGCGTCATCGGGGCTTAGGGAGATCCTCCTACTGGG
TTTCGGGTATATTGGCCCACATCCAGCGAACCATCG
TGACGTTGATAGGGTTCCATACGGAGCCAACAATGT
GGGCGGATATGTTACACCGAACAGTGGTAATAAGGG.

El paso 6, aplica la regla de sustracción de ADN $L_2 = K - DADN$.

L_2 = CCCACAGCCGTTCTAGCGTGCTCACGTTCCCACGGC
CGGACGTGCGTGCGCCCTCAAGAACTAGCGCCCGCC
CGCCCTCACGACCTATCGCCCGACCGTCCGATCTAT
AGAACGCCCGTACTATCGTACGTGCGACCGGAAGAA
ATACCTAGAGAAAGAACGTACGCACTCACGTGATAG
ATAACGTGCAATCGTTCGGCAGAACGGCCGTTATAA
CTCACGACCGTTCGTGCGCTCAGCCGTTCGTAATAC
CGGACTCACGTGAGAACGCCCGTGCGTGCGTTATCG
AGAACGGCCGCTCAGCCGTGCGCGAGAACGCTAGCC.

El paso 7, pasa L_2 a binario y el paso 8 pasa a valores decimales.

DI_d =
84 73 111 114 110 116 111 84 105
104 110 110 101 116 32 114 101 101
101 116 97 115 101 97 109 99 115
32 101 108 115 108 110 97 104 32
49 114 32 32 108 100 116 110 50
48 110 67 111 105 32 105 111 48
116 97 111 110 103 73 111 108 49
104 116 110 32 101 110 110 111 54
32 105 103 73 110 102 32 103 37.

El paso 9, pasa la matriz DI_d a un vector desapilando por columnas y el paso 10 elimina la basura obteniendo el vector:

DT_d =
84 104 101 32 49 48 116 104 32 73 110 116 101 114
110 97 116 105 111 110 97 108 32 67 111 110 103 114
101 115 115 32 111 110 32 73 110 116 101 108 108 105
103 101 110 116 32 97 110 100 32 73 110 102 111 114
109 97 116 105 111 110 32 84 101 99 104 110 111 108
111 103 105 101 115 32 50 48 49 54.

El paso 11, pasa DT_d a caracteres:
T = "*The 10th International Congress on Intelligent and Information Technologies 2016"*.

4.2. Seguridad del cripto sistema

En esta sección se prueba la fortaleza del sistema criptográfico propuesto.

4.2.1. Ataque a fuerza bruta

El primer análisis que se presenta es el de ataque por fuerza bruta, que consiste en probar todas las posibles llaves hasta encontrar la llave con la cual se pueda recuperar el mensaje original. Para considerar que el espacio de llaves es adecuado para resistir un ataque de fuerza bruta, este espacio debe ser superior a $2^{100} = 1.2677 \times 10^{30}$ [12]. El sistema criptográfico que aquí se presenta, depende de 2 llaves, la primera generada a partir de la secuencia caótica dada por las ecuaciones (2) y (3) dependiendo de dos números aleatorios iniciales x_0, y_0 en el intervalo real [0,1], al usar la función con menos

rango de posibles números aleatorios de Octave se tiene un total de 2^{32}-1 posibilidades para cada número, la segunda llave también depende de dos valores *a* y *b*, por lo que el total de posibles llaves es: (2^{32}-1) x (2^{32}-1) x (2^{32}-1) x (2^{32}-1) = 3.4028×10^{38}, dando seguridad al algoritmo criptográfico propuesto en este trabajo ante el ataque a fuerza bruta.

4.2.2 Sensibilidad de los datos de entrada

Los sistemas caóticos tienen la característica de ser sensibles a las condiciones iniciales [14]. Es por ello, que el segundo análisis que se presenta es el de sensibilidad de los datos de entrada a un cambio muy pequeño en los valores iniciales de la generación de llaves.

Para la cross chaotic map [18] usada en esta propuesta, la sensibilidad se ilustra con un ejemplo, realizando la ejecución de la implementación computacional con los valores iniciales $x_0 = 0.3$, $y_0 = 0.6$ dejando sin cambio los valores iniciales de la segunda llave, se descifra de forma correcta el mensaje dado por la ecuación (8), pero al usar los valores iniciales modificados de $x_0 = 0.30001$ y $y_0 = 0.5999$ se obtiene el mensaje descifrado " *Wkd!21wk#mBXA^JEXM jila%Djib=«X6ø- ------- ä Ñþô°ùÞÖ[nYuH}[Z-³µ¶ìèö-¿½´¹"* que claramente no corresponde al mensaje original de la ecuación (8).

También, dejando fijos los valores iniciales de la función caótica y con valores de $a=0.3$ y $b=0.6$ en la segunda llave se obtiene el mensaje original de la ecuación (8), sin embargo, modificando ligeramente los valores iniciales *a* y *b* como 0.30001 y 0.59999 se obtiene el descifrado " *Uif#03ui!eJPIVBMPE`cfk/N`chª « ø7X±- Ä°ñÞÔ ÙþösFq]`Usr4¯¼º·¶¾æâü§µ·¾³* " que nuevamente no corresponde al mensaje original. Por lo tanto, se puede decir, que el sistema criptográfico es muy sensible a variaciones pequeñas en las llaves brindando una gran resistencia ante ataques exhaustivos.

5. Conclusiones

El uso de las funciones caóticas, junto con las operaciones de adición y sustracción sobre ADN y las raíces de funciones no lineales son útiles en el desarrollo de sistemas criptográficos simétricos. Nuestra propuesta usa dos llaves, la primera se genera a partir de las funciones caóticas como lo hacen los autores que se mencionan, la segunda se genera a partir de las constantes de una función no lineal, dándole por consiguiente más seguridad al método criptográfico y sólo se incrementa en dos datos más (las constantes reales *a* y *b* usadas en la función no lineal dada por la ecuación 7) los elementos usados como llaves.

El sistema criptográfico presentado es muy sensible como se muestra en la sección 4.2 a variaciones pequeñas en las llaves dando por consiguiente gran resistencia ante ataques exhaustivos.

Agradecimientos. Agradecemos el apoyo financiero de la Vicerrectoría de Investigación y Estudios de Posgrado de la Benemérita Universidad Autónoma de

Puebla a través del proyecto MACL-ING16-I, así como a los revisores de este trabajo por sus comentarios constructivos.

Referencias

1. Anil, J., Chirakkarott, S.: Secure Encryption Method for Biometric Iris Pattern. International Journal of Trends and Technology (IJCTT), Vol. 12, No. 5, pp. 217–219 (2014)
2. Anwar, T., Sanchita, P., Singh, S. K.: Message Transmission Based on DNA Cryptography: Review. International Journal of Bio-Science and Bio-Technology, Vol. 6, No. 5, pp. 215–222 (2014)
3. Burden, R., Douglas, F. J.: Numerical methods. 4th edition, International edition (2012)
4. Delfs, H., Maurer, U., Knebl, H.: Introduction to Cryptography Principles and Applications. Second Edition, Springer-Verlag Berlin Heidelberg (2007)
5. Fuan, Ch., Mengb, W., Zhanb, Y., Zhuc, Z., Laud, F., Tsed, Ch., Mae, H.: An efficient and secure medical image protection scheme based on chaotic maps. Computers in Biology and Medicine, Vol. 43, pp. 1000–1011 (2013)
6. Galende, J.: Criptografía, historia de la escritura cifrada. 1ra edición, editorial complutense (1995)
7. Gao, T. G., Chen, Z. Q.: Image encryption based on a new total shuffling algorithm. Chaos Solutions & Fractals, 38(1), pp. 213–220 (2008)
8. Goldreich, O.: Modern Cryptography, theory and practice, discrete mathematics and its applications. 3th edition, Chapmann & Hall (1999)
9. Hilborn, R. C.: Chaos and nonlinear dynamics. 2da edition, Oxford University Press (2000)
10. Huang, C. K., Nien, H. H.: Multi chaotic systems based pixel shuffle for image encryption. Opt. Commun., 282(11), pp. 2123–2127 (2009)
11. Inzunza, E., Cruz, C.: Double hyperchaotic encryption for security in biometric systems. Nonlinear Dynamics and Systems Theory, 13(1), pp. 5–68 (2013)
12. Jeevidha, S., Saleem, M. S., Dhavachelan, P.: Analysis on DNA based cryptography to secure data transmission. International Journal of International Journal of Computer Applications, Vol. 29, No. 8 (2011)
13. Javheri, S., Kulkarni, R.: Secure Data communication and Cryptography based on DNA based Message Encoding. International Journal of Computer Applications, Vol. 98, No. 16, pp. 35–40 (2014)
14. Ott, E.: Chaos in dynamical systems. Cambridge University Press (1993)
15. Patidar, V., Pareek, N. K., Sud, K. K.: A new substitution–diffusion based image cipher using chaotic standard and logistic maps. Commun. Nonlinear Sci. Numer. Simulation, 14(7), pp. 3056–3075 (2009)
16. Rhouma, R., Meherzi, S., Belghith, S.: OCML-based colour image encryption. Chaos Solitons & Fractals, 40(1), pp. 309–318 (2009)
17. Sun, F. Y., Liu, S. T., Li, Z. Q.: A novel image encryption scheme based on spatial chaos map. Chaos Solitons & Fractals, 38(3), pp. 631–640 (2008)
18. Teree, R., Vaida, M. F., Alboaie, L., Chiorea, L.: DNA security using symmetric cryptography. International journal of new computer architectures and their applications, IJNCAA, Vol. 1 (2011)
19. Tong, X. J., Cui, M. G.: Image encryption scheme based on 3D baker with dynamical compound chaotic sequence cipher generator. Signal Processing, 89(4) pp. 480–491 (2009)
20. Wong, K. W., Kwok, B. S., Law, W. S.: A fast image encryption scheme based on chaotic standard map. Phys. Lett. A., 372(15), pp. 2645–2652 (2008)
21. Xiao, Y. L., Xia, L. M.: An Image Encryption Approach Using a Shuffling Map. Commun. Theor. Phys, 52(5), pp. 876–880 (2009)

22. Xu, S. J., Wang, J. Z., Yang, S. X.: An improved image encryption algorithm based on chaotic maps. Chin. Phys. B, 17(11), pp. 4027–4032 (2008)
23. Ye, G. D.: A chaotic image cryptosystem based on Toeplitz and Hankel matrices. Imaging Sci. J., 57(5), pp. 266–273 (2009)
24. Octave: Disponible en https: //www.gnu.org/software/octave (2016)