Turkish Journal of Science & Technology
18(1), 261-273, 2023

Research Paper
https://doi.org/10.55525/tjst.1250419

CDIEA: Chaos and DNA Based Image Encryption Algorithm

Ali ARI[1*],
[1] Scientific and Technological Research Center, Inonü University, Malatya, Türkiye
[*1] ali.ari@inonu.edu.tr

(Geliş/Received: 16/02/2023; Kabul/Accepted: 21/03/2023)

Abstract: A proposal for an image encryption algorithm called Chaos and DNA Based Image Encryption Algorithm (CDIEA) has been put forward. CDIEA is a combination of block cipher algorithms, permutations, chaotic keys, and DNA operations. It leverages the strong structures of modern cryptography and the properties of chaotic systems, rather than relying solely on chaos. The permutations used in CDIEA are constructed using a strategy called the wide trail design, which makes it resilient to many forms of cryptanalysis. CDIEA operates as a byte-oriented SP-network and has been confirmed to have high security for practical image encryption through both theoretical analysis and computer experiments.

Key words: Image encryption; Chaos; DNA; Cryptography

CDIEA: Kaos ve DNA Tabanlı Görüntü Şifreleme Algoritması

Öz: Kaos ve DNA Tabanlı Görüntü Şifreleme Algoritması (CDIEA) adı verilen bir görüntü şifreleme algoritması önerisi ortaya atılmıştır. CDIEA, blok şifreleme algoritmaları, permütasyonlar, kaotik anahtarlar ve DNA işlemlerinin bir kombinasyonudur. Yalnızca kaosa güvenmek yerine, modern kriptografinin güçlü yapılarından ve kaotik sistemlerin özelliklerinden yararlanır. CDIEA'da kullanılan permütasyonlar, onu birçok kriptanalize dayanıklı hale getiren geniş iz tasarımı adı verilen bir strateji kullanılarak oluşturulmuştur. CDIEA, bayt yönelimli bir SP ağı olarak çalışır ve hem teorik analiz hem de bilgisayar deneyleri yoluyla pratik görüntü şifreleme için yüksek güvenliğe sahip olduğu onaylanmıştır.

Anahtar kelimeler: Görüntü şifreleme, Kaos, DNA, Kriptoloji

1. Introduction

The importance of communication security arises with the rapid developments in computer and communication technologies. Many mathematicians, computer scientists and electrical engineers that are professional in their fields focused on the subject of cryptology to provide the security of millions of users who process in electronic environments. Researchers have been proposed several cryptographic algorithm and protocol using abstract algebra, number theory, and coding theory [1, 2]. Cryptology experts are constantly working to develop more powerful, faster, robust, and reliable encryption systems. Chaos based cryptology is one of the new methods introduced to carry out this goal [3-6]. Utilization of chaotic systems in the process of designing cryptographic structures is theoretically feasible since the characteristics of chaotic systems overlap with those of the cryptographic characteristics [3, 4]. Many studies have been done for design and analysis of chaos based encryption systems over the last 20 years. One of the most common subjects studied in this field is chaos based image encryption [7-13]. Nevertheless, the cryptanalysis of these algorithms has shown several significant weaknesses [14-20]. The performed analysis studies have shown that the encryption systems determined by weak transformations are not secure. One specific type of attack against chaos-based encryption targets the chaotic aspect of the cryptography. The encryption process is transformed into an alternate form where the chaotic

[*] Corresponding author: ali.ari@inonu.edu.tr ORCID Number of author: [1] 0000-0002-5071-6790

components are replaced by secret mapping or variables. This exposes any algebraic vulnerabilities in the remaining parts of the algorithm [5, 14, 15]. In this research, a novel image encryption algorithm called CDIEA has been proposed. It merges the robust structures of modern cryptography with the principles of chaos and DNA operations. CDIEA was created as a solution to the inadequacy of traditional text encryption methods in protecting image data due to its unique properties. The algorithm is an iterated block cipher based on permutations, chaotic keys with uniform distribution, and DNA operations, and operates as a byte-oriented SP-network. The encryption scheme's transformations were modeled after those of the AES block cipher, leading to exceptional confusion and diffusion capabilities.

The paper is structured as follows. The CDIEA method is explained in detail in Section 2, where its unique features and design rationale are outlined. Section 3 showcases the early results of our security analysis of CDIEA. Finally, the paper concludes in Section 4.

2. Structure of CDIEA construction

Main modules of CDIEA are illustrated in Figure 1. CDIEA consist of four main modules. These modules are DNA operations module, chaotic key generation module, encryption module, and chaotic bit block permutation module. The algorithm’s operation is given step-by-step below.

Step 1. Plain image is converted into a one-dimensional bit array.

Step 2. The one-dimensional bit array M split into 512-bit blocks $m_1, m_2, \ldots, m_l$,

Step 3. DNA operations are applied for each bit block. In Section 2.1, DNA operations are described in detail.

Step 4. Chaotic key generator module produces a 512-bit secret key for each block. In Section 2.2, chaotic key generator module is described in detail.

Step 5. The encryption component transforms two inputs, each consisting of 512 bits, into an output of 512 bits. The first input is the chaotic key input and the second is the bit block. The details of the encryption module can be found in Section 2.3.

Step 6. Ciphered bit blocks are permutated using chaotic permutation module. In Section 2.4, this module is described.

Step 7. Steps 3-6 are performed 10 times for each 512-bit block.

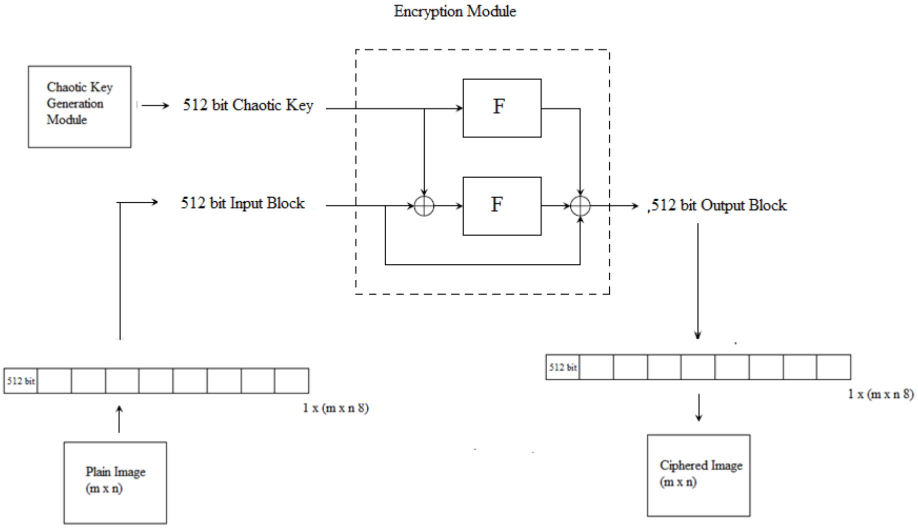

Figure 1. Main modules of CDIEA

2.1. DNA operations module

DNA computing is a novel approach to computing that employs DNA instead of conventional methods. It is a multidisciplinary field that is rapidly advancing and has yielded a number of biological and algebraic operations based on DNA sequences [23-25]. A DNA sequence consists of four base pairs, namely A, C, G, and T, where A and T are complementary to each other, as well as C and G. Information is represented by DNA sequences in DNA computing, and binary numbers are used to represent the four bases. Each base is represented by two bits of binary information. In the binary system, 0 and 1 are complementary, and similarly, the combinations 00 and 11, as well as 01 and 10, are also complementary. The four bases can be expressed by 00, 01, 10, and 11, resulting in 24 possible coding combinations. However, due to the complementary relationship between DNA bases, only eight coding combinations that satisfy this principle exist. Table 1 lists these eight encoding rules. For instance, the binary pixel value of an image is [10010011], which corresponds to the DNA sequence [GCAT] based on the first encoding rule. Similarly, the decoding sequence can be determined as [00111001] according to the sixth decoding rule.

Table 1. DNA coding rules

Combination	Coding Rule
Rule Combination 1	A: 00 C: 01 G: 10 T: 11
Rule Combination 2	A: 00 G: 01 C: 10 T: 11
Rule Combination 3	C: 00 A: 01 T: 10 G: 11
Rule Combination 4	C: 00 T: 01 A: 10 G: 11
Rule Combination 5	G: 00 A: 01 T: 10 C: 11
Rule Combination 6	G: 00 T: 01 A: 10 C: 11
Rule Combination 7	T: 00 C: 01 G: 10 A: 11
Rule Combination 8	T: 00 G: 01 C: 10 A: 11

In the DNA operation module, 512-bit output block are obtained by applying the following operations for each 512-bit input blocks.

- DNA encoding rule selected by using key$_1$ (key$_1$ is part of the secret key. $key_1 \in [1,8]$) and get DNA sequence matrixes D$_1$.
- DNA encoding rule selected by using key$_2$ (key$_2$ is part of the secret key. $key_2 \in [1,8]$ and $key_1 \neq key_2$) and get DNA sequence matrixes D$_2$.
- Carry out DNA addition operation DNA sequence matrixes. DNA addition and subtraction rules are given Table 2.
- DNA decoding rule selected by using key$_3$ (key$_3$ is part of the secret key. $key_3 \in [1,8]$) and get binary bit sequence.

Table 2. DNA addition and subtraction rules

+/-	**A**	**T**	**C**	**G**
G	C	A	G	G
C	A	T	C	T
T	G	C	T	A
A	T	G	A	C

2.2. Chaotic Key Generation Module

Chaotic systems are highly influenced by their starting conditions and control parameters. The path of chaotic signals exhibit unpredictability and random-like behavior. Mathematically, chaos is the randomness in a straightforward deterministic dynamic system [26]. Chaos-generated random numbers have also been employed as secret keys [27-30]. The utilization of chaotic numbers is backed by their uncertainty, wide-spectrum characteristics, non-repeating, intricate time-based behavior, and ergodic qualities.This module is used to generate round keys. Algorithm is explained in detail below.

Step 1: Choose a chaotic system to serve as the source of randomness.

Step 2: Determine the state variables based on the selected initial conditions and control parameters.

Step 3: Using Eq. (1), generate pseudorandom numbers using the selected state variable.

ChaoticKey(i)=(1000*(x(i)+abs(x(i))))%1 (1)

System equations, initial conditions, and control parameters of some chaotic systems are given in Table 3 [26].

Table 3. Examples of chaotic systems

Chaotic Systems	**Equation**	**Initial Values**	**Control Parameters**
Chen system	$\begin{cases} \dot{x} = a(y-x) \\ \dot{y} = (c-a)x - xz + cy \\ \dot{z} = xy - bz \end{cases}$	(x_0, y_0, z_0) $= (-3, 2, 20)$	$a = 35, b =$ $3,$ and $c = 28$
Chua circuit	$\frac{dV_{C1}}{dt} = \frac{1}{C_1 R}(V_{C2} - V_{C1}) - f(V_{C1})$ $\frac{dV_{C2}}{dt} = \frac{1}{C_2 R}(V_{C1} - V_{C2}) - i_L$ $\frac{di_L}{dt} = -\frac{V_{C2}}{L}$	$1/L = 25.58$	$1/C_1 R = 15.6$, $1/C_2 R = 1$, $a = -5/7$ and $b = -8/7$
Lorenz system	$\frac{dx}{dt} = a(y-x)$ $\frac{dy}{dt} = (bx - y - xz)$ $\frac{dz}{dt} = (xy - cz)$	-20≤x≤20, 50≤y≤50, 50≤z≤50	a=10, b=28 and c=8/3
Van der Pol oscillator	$dx = y$ $dy = a*(1-x^2)*y - x^3 + b*\cos(c*z)$ $dz = 1$	$x \;\; = 0$ $y_0 = 0$ $z_0 = 0$	a=0.2, b=5.8 and c=3

2.3. Encryption Module

The structure of encryption module is illustrated in Figure 2. Encryption module uses a permutation. Three transformations are defined for each permutation. Transformations are illustrated in Figure 3. These are substitution, shift, and mix transformations. The transformations operate on a state, which is represented as a matrix *A* of bytes (of 8 bits each). The matrix has 8 rows and 8 columns.

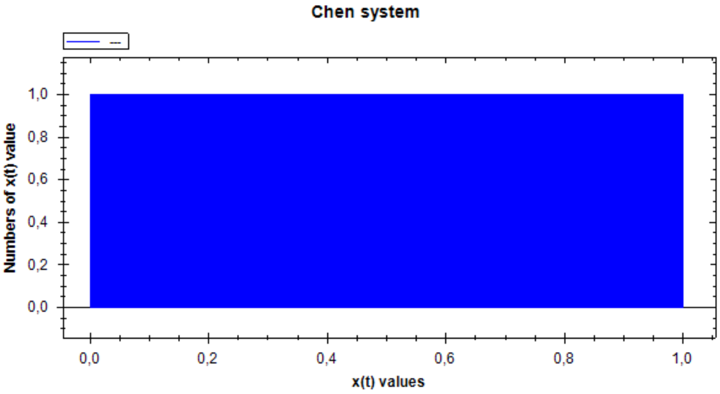

Figure 2. Distribution diagram analysis for Chen system

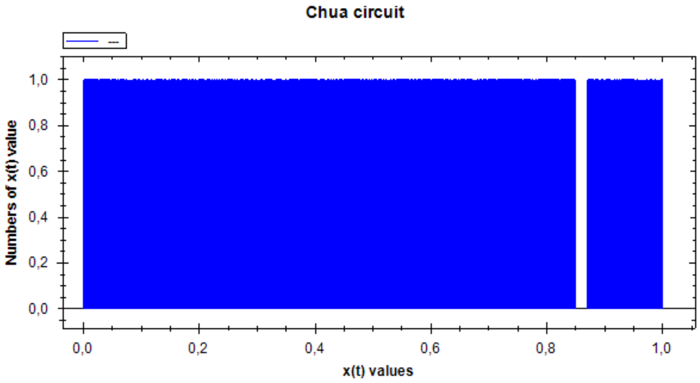

Figure 3. Distribution diagram analysis for Chua circuit

The substitution transformation substitutes each byte in the state matrix by another value, taken from the s-box. This s-box is the same as the one used in AES. If $a_{i;j}$ is the element in row *i* and column *j* of A, then substitution performs the Eq. (2).

$$a_{i,j} \leftarrow S(a_{i,j}) \qquad (2)$$

Shift transformation cyclically shifts the bytes within a row to the left by a number of positions. The shift transformation is illustrated in Figure 4.

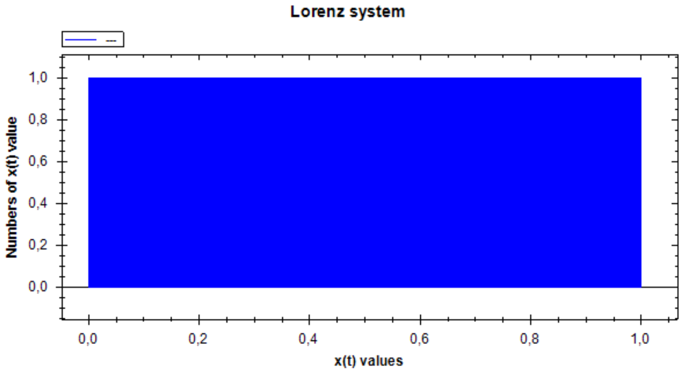

Figure 4. Distribution diagram analysis for Lorenz system

The mix transformation operates on each column of the matrix independently. The elements of the state matrix A are considered to be bytes in F256. This transformation involves multiplying each column of A by a constant 8x8 matrix B in F256. The overall transformation of the matrix A can be represented by Eq. (3).

$$A \leftarrow B \times A \tag{3}$$

The matrix *B* is specified via the irreducible polynomial $x^8 \oplus x^4 \oplus x^3 + x + 1$ over F$_2$. *B* can be written as Eq. (4).

$$B = \begin{bmatrix} 02 & 02 & 03 & 04 & 05 & 03 & 05 & 07 \\ 07 & 02 & 02 & 03 & 04 & 05 & 03 & 05 \\ 05 & 07 & 02 & 02 & 03 & 04 & 05 & 03 \\ 03 & 05 & 07 & 02 & 02 & 03 & 04 & 05 \\ 05 & 03 & 05 & 07 & 02 & 02 & 03 & 04 \\ 04 & 05 & 03 & 05 & 07 & 02 & 02 & 03 \\ 03 & 04 & 05 & 03 & 05 & 07 & 02 & 02 \\ 02 & 03 & 04 & 05 & 03 & 05 & 07 & 02 \end{bmatrix} \tag{4}$$

2.4. Chaotic Permutation Module

CDIEA is a block encryption algorithm with 512-bit block size. A grey-level image P(m, n), where m, n are the image dimensionalities of rows and columns, has contain $l = (m \times n \times 8)/512$ blocks. Permutation matrix is generated using the following algorithm.

Step 1: Choose a chaotic system as the source of randomness.

Step 2: Obtain the state variables based on selected initial conditions and control parameters.

Step 3: Collect samples of the selected state variable at regular intervals equal to (number of data / l).

Step 4: Assign numerical codes to each sample starting from 0 to l.

Step 5: Create a permutation matrix using the codes assigned to the samples, with the code of the smallest output being placed in the first cell.

Step 6: Apply the permutation matrix to the ciphered blocks to replace them.

In order to illustrate how permutation matrix is generated with an example a simple system orbit is given in Figure 5. By taking 16 samples on the orbit a 1x16 permutation matrix is generated. Output values, the corresponding code value for each sample taken on the orbit, and the formed permutation matrix are given in Table 4. (For details see [31])

Table 4. Example of chaotic permutation

Sampling Order	**No 1**	**No 2**	**No 3**	**No 4**	**No 5**	**No 6**	**No 7**	**No 8**	**No 9**	**No 10**	**No 11**	**No 12**	**No 13**	**No 14**	**no 15**	**No 16**
Output Value	30.2	8.9	1.7	3.4	14.2	0.01	5,6	19,6	28.2	16.9	23.7	32.4	25.2	6.01	9,6	7
First Codes	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Permutation Matrix	5	2	3	6	13	15	1	14	4	9	7	10	12	8	0	11

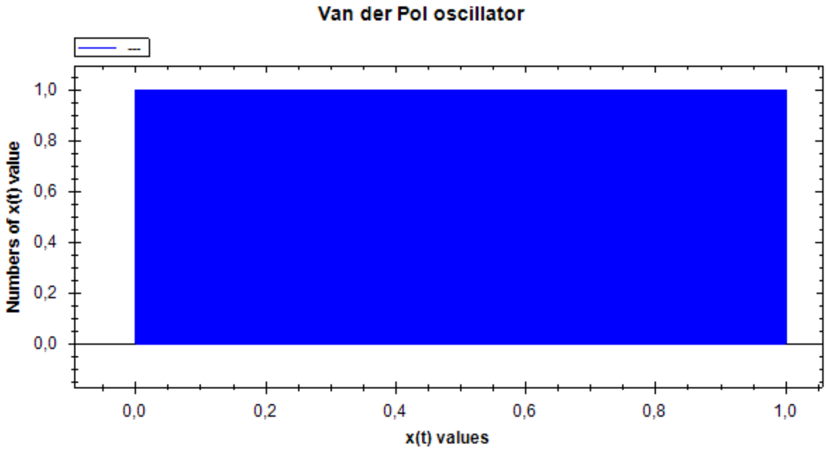

Figure 5. Distribution diagram analysis for Van der Pol oscillator

3. Security Analysis

The protection of a cryptographic system is of utmost importance. A secure encryption scheme should be able to defend against various forms of attacks, including cryptanalytic attacks, such as statistical attacks, brute-force attacks, attacks based solely on ciphertext, and attacks with access to known plaintext, among others [32-35]. In this section, the results of a security analysis of CDIEA will be explored.

3.1. Differential cryptanalysis

The permutation process in CDIEA is designed to have diffusion properties, meaning it helps to scramble the data and make it more difficult for an attacker to reverse the encryption. This is due to the combination of the mix transformation having nine branches and the shift transformation being optimized to move the data in each column to eight different columns. As a result, CDIEA is expected to have at least 81 active s-boxes in any four-round differential trail. This, in combination with the s-box having a maximum difference propagation probability of 2^{-6}, means that the probability of any differential trail being successful over four rounds is estimated to be at most 2^{-486}. So, in a classical differential attack scenario where an attacker tries to find a specific pattern in every round, it is highly unlikely they will succeed in breaking CDIEA's encryption.

3.2. Linear cryptanalysis

The propagation of linear and differential trails is quite alike [35]. As the mix transformation has a linear branch number of 9, it is guaranteed that in any four-round linear trail, CDIEA will have at least 81 active s-boxes [22, Theorem 9.5.1]. Given that the s-box has a maximum correlation of 2^{-3}, the highest correlation for a four-round linear trial is estimated to be $2^{-3 * 81} = 2^{-243}$.

3.3. Algebraic cryptanalysis

It is well established that 40 quadratic equations can be established from the input and output bits of the AES s-box [21, 22]. These equations, along with one additional equation with a probability of 255/256, also apply to the s-box used in CDIEA. Using these equations, it has been demonstrated that a single AES encryption can produce a set of 8000 quadratic equations in 1600 variables, which can then be used to determine the secret key. Although the time complexity of solving this system of equations for AES is currently unknown, it is believed that it would take longer than an exhaustive search for the key. In comparison, the encryption function of CDIEA utilizes 1280 s-box applications, making it less vulnerable to algebraic attacks than AES. However, if an efficient attack method is discovered for CDIEA that exploits the quadratic s-box equations, it is likely that a similar attack would also be successful for AES.

3.4. Security analysis of chaotic key generation module

The performance of pseudo-random number generators (PRNGs) must be statistically indistinguishable from truly random sequences. Evaluating the quality of PRNGs requires statistical analysis of their output sequences by using statistical randomness tests. These tests are designed to test a null hypothesis (H0), which states that the input sequence is random. The test takes a binary sequence as an input and decides to either accept or reject the hypothesis. Statistical tests have a probabilistic nature and there are two possible types of errors. A type I error occurs when the data is random but the null hypothesis is rejected. A type II error occurs when the data is non-random but the null hypothesis is accepted. The probability of a type I error is referred to as the level of significance of the test, represented by alpha (α). A statistical test produces a p-value between 0 and 1, where if the p-value is greater than alpha, the null hypothesis is accepted; otherwise, it is rejected.

A test suite is a compilation of statistical randomness tests that aim to assess the randomness qualities of sequences. There are a number of test suites in the literature, including:

- The first set of randomness tests by Knuth, which was introduced in his well-known book [36].
- CRYPT-X, developed at the Queensland University of Technology [37].
- The DIEHARD Test Suite, created by Marsaglia and released in 1995 on CDROM [38].
- TESTU01, a newly designed test suite which categorizes tests into those for real numbers in (0, 1) and those for bits [39].
- The NIST Test Suite, which initially consisted of 15 tests [40].

NIST Test Suite has been used in this study to assess randomness. The obtained results are shown in Table 5. The names of the NIST tests listed in Table 5 are Test 1: Monobit Test, Test 2: Frequency Within Block Test, Test 3: Runs Test, Test 4: Longest Run Ones in a Block Test, Test 5: Binary Matrix Rank Test, Test 6: Dicrete Fourier Transform Test, Test 7: Non Overlapping Template Matching, Test 8: Overlapping Template Matching, Test 9: Maurers Universal Test, Test 10: Linear Complexity Test, Test 11: Serial test, Test 12: Approximate Entropy Test, Test 13: Cumulative Sums Test, Test 14: Random Excursion Test, and Test 15: Random excursion variant test, respectively.

Table 5. Results of SP 800-22 test

Test Name	Chen system	Chua circuit	Lorenz system	Van der Pol oscillator
Test 1:	√	√	√	√
Test 2:	√	√	√	√
Test 3:	√	√	√	√
Test 4:	√	√	√	√
Test 5:	√	√	√	√
Test 6:	√	√	√	√
Test 7:	√	√	√	√
Test 8:	√	√	√	√
Test 9:	√	√	√	√
Test 10:	√	√	√	√
Test 11:	√	√	√	√
Test 12:	√	√	√	√
Test 13:	√	√	√	√
Test 14:	√	√	√	√
Test 15:	√	√	√	√

4. Conclusion

A new image encryption method called CDIEA has been introduced. It uses chaos and DNA encoding to create permutations, which are based on parts of the AES block cipher, providing CDIEA with superior diffusion and confusion characteristics. The design of CDIEA is straightforward, making it a robust image encryption algorithm that can be easily implemented on various platforms.

References

[1] Katz J, Lindell Y. Introduction to modern cryptography: principles and protocols, Chapman & Hall, 2008.

[2] Paar C, Pelzl J. Understanding Cryptography a Textbook for Student and Practitioners, Springer, 2010.

[3] Amigo JM, Kocarev L, Szczapanski J. Theory and practice of chaotic cryptography, Phys Lett A 2007; 366: 211-216.

[4] Alvarez G, Li S. Some basic cryptographic requirements for chaos-based cryptosystems. Int J Bifurcat Chaos 2006;16 (8): 2129–2151.

[5] Solak E. Cryptanalysis of Chaotic Ciphers, in: L. Kocarev, S. Lian (Eds.), Chaos Based Cryptography Theory Algorithms and Applications, Springer-Verlag 2011; 227-256.

[6] Alvarez G, Amigo JM, Arroyo D, Li S. Lessons Learnt from the Cryptanalysis of Chaos-Based Ciphers, in: L. Kocarev, S. Lian (Eds.), Chaos Based Cryptography Theory Algorithms and Applications, Springer-Verlag 2011; 257-295.

[7] Fridrich J. Symmetric ciphers based on two-dimensional chaotic maps, Int J Bifurcat Chaos 1998; 8(6): 1259–1284.

[8] Patidar V, Pareek NK, Sud KK. A new substitution-diffusion based image cipher using chaotic standard and logistic maps. Commun Nonlinear Sci Numer Simul 2009; 14(7): 3056–3075.

[9] Zhu C. A novel image encryption scheme based on improved hyperchaotic sequences, Opt Commun 2012; 285(1): 29-37.

[10] El-Latif A, Li L, Wang N, Han Q, Niu X. A new approach to chaotic image encryption based on quantum chaotic system, exploiting color spaces, Signal Processing, 2013; 93(11): 2986-3000.

[11] Zhang Q, Guo L, Wei X. Image encryption using DNA addition combining with chaotic maps, Math Comput Model 2010; 52: 2028-2035.

[12] Bigdeli N, Farid Y, Afshar K. A robust hybrid method for image encryption based on Hopfield neural network, Computers and Electrical Engineering 2012; 38: 356–369.

[13] Liu L, Zhang Q, Wei X. A RGB image encryption algorithm based on DNA encoding and chaos map, Computers & Electrical Engineering 2012; 38(5):1240-1248.

[14] Arroyo D, Diaz J, Rodriguez FB. Cryptanalysis of a one round chaos-based Substitution Permutation Network, Signal Processing 2013; 93(5): 1358-1364.

[15] Özkaynak F, Özer AB, Yavuz S. Cryptanalysis of a novel image encryption scheme based on improved hyperchaotic sequences, Opt Commun 2012; 285: 4946–4948.

[16] Özkaynak F, Özer AB, Yavuz S. Cryptanalysis of Bigdeli algorithm using Çokal and Solak attack, International Journal of Information Security Science 2012; 1(3): 79-81.

[17] Özkaynak F, Özer AB, Yavuz S. Analysis of Chaotic Methods for Compression and Encryption Processes in Data Communication, 20th IEEE Signal Processing and Communications Applications Conference 2012.

[18] Özkaynak F, Özer AB, Yavuz S. Security Analysis of an Image Encryption Algorithm Based on Chaos and DNA Encoding, 21th IEEE Signal Processing and Communications Applications Conference 2013.

[19] Solak E, Çokal C, Yildiz OT, Biyikoglu T. Cryptanalysis of fridrich's chaotic image encryption. Int J Bifurcat Chaos 2010; 20(5): 1405–1413.

[20] Rhouma R, Solak E, Belghith S. Cryptanalysis of a new substitution–diffusion based image cipher, Commun Nonlinear Sci Numer Simul 2010; 15(7) : 1887-1892.

[21] Knudsen L, Robshaw M. The Block Cipher Companion, Springer, 2011.

[22] Daemen J, Rijmen V. AES Proposal: Rijndael, First Advanced Encryption Conference, California, 1998.

[23] Gaborit P, King OD. Linear constructions for DNA codes, Theor Comput Sci 2005; 334: 99–113.

[24] Xiao GZ, Lu MX, Qin L, Lai XJ. New field of cryptography: DNA cryptography, Chin Sci Bull 2006; 51(12): 1413–1420.

[25] Gehani A, LaBean TH, Reif JH. DNA-based cryptography. DIMACS series in discrete mathematics, Theor Comput Sci 2000; 54: 233–249.

[26] Sprott J. Elegant Chaos Algebraically Simple Chaotic Flows. World Scientific, 2010.

[27] Stojanovski T, Kocarev L. Chaos-based random number generators – Part I: Analysis, IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications 2001; 48: 281-288.

[28] Kocarev L, Jakimoski G. Pseudorandom bits generated by chaotic maps, IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications 2003; 50: 123-126.

[29] Barash L, Shchur LN. Periodic orbits of the ensemble of Sinai-Arnold cat maps and pseudorandom number generation, Phys Rev E 2006; 73: 036701.

[30] Barash LY, Shchur LN. RNGSSELIB: Program library for random number generation, SSE2 realization, Comput Phys Commun 2011; 182(7):1518-1527.

[31] Özkaynak F, Özer AB. A method for designing strong S-Boxes based on chaotic Lorenz system. Phys Lett A 2010; 374: 3733–3738.

[32] Bard GV. Algebraic Cryptanalysis, Springer, 2009.

[33] Joux A. Algorithmic cryptanalysis, Chapman & Hall, 2009.
[34] Biham E, Shamir A. Differential Cryptanalysis of DES-like Cryptosystems, Journal of Cryptology 4 1991; 3-72.
[35] Matsui M. Linear Cryptanalysis Method for DES Cipher, Advances in Cryptology - Eurocrypt '93, Lecture Notes in Computer Science 1994; 765: 386-397.
[36] Knuth DE. Seminumerical Algorithms, volume 2 of The Art of Computer Programming, Addison-Wesley, 1981.
[37] Caelli W, Dawson E, Nielsen L, Gustafson H. CRYPT–X statistical package manual, measuring the strength of stream and block ciphers, 1992.
[38] Marsaglia G. The Marsaglia random number CDROM including the DIEHARD battery of tests of randomness, 1996.
[39] L'Ecuyer P, Simard R. Testu01: A c library for empirical testing of random number generators. ACM Trans Math Softw 2007; 33(4): 22.
[40] Rukhin A, Soto J, Nechvatal J, Smid M, Barker E, Leigh S, Levenson M, Vangel M, Banks D, Heckert A, Dray J, Vo S. A statistical test suite for random and pseudorandom number generators for cryptographic applications, 2001.