Nonlinear Phenomena in Complex Systems, vol. **19**, no. 4 (2016), pp. 403 - 410

Block Encryption Algorithm Based on Dynamic Chaos

A. V. Sidorenko*
Belarusian State University, 4 Nezavisimosti Ave., Minsk 220030, BELARUS
(Received 18 January, 2016)

An algorithm using dynamic chaos for encryption and decryption of the plain text has been developed. To prove the stability of the proposed encryption algorithm, the quantitative parameters have been determined: information entropy, numerical characteristics for the distribution of byte values in open and encrypted texts, correlation coefficient, percentage of the value-changing bits (avalanche effect), number of pixels change rate (NPCR), unified average changing intensity (UACI).

PACS numbers: 05.45.Gg, 05.45.Vx
Keywords: cryptanalysis, differential, linear, algorithm, encryption, Feistel cipher, dynamic chaos, security

1. Introduction

In contemporary world information resources are most powerful levers of progress. Along with the traditional encryption algorithms which are constantly developed, improved, and upgraded, the encryption algorithms based on dynamic chaos systems become increasingly popular with cryptographic community.

Systems of dynamic chaos are dynamic systems with an exponential dependence of the state on initial conditions: a small variation in an initial state of a system leads to substantial changes in its entire trajectory in the phase space. A change in the initial conditions is exponentially amplified in time.

The goal of this paper is to develop an encryption algorithm and software based on dynamic chaos, and to examine its resistance to various types of cryptographic attacks.

2. Formation principle of block ciphers using chaotic maps

In block encryption and synthesis of digital chaotic ciphers a plain text (and/or a secret key) is used as initial conditions and/or control parameters for consecutive iterations of a chaotic system.

As a rule, the alphabet for a block cipher represents a number of binary vectors that are blocks of plain text having the same length. Characteristics for such a block cipher is its ability to encrypt with one key one or more messages, their total length exceeding a length of the key.

Sending of a key, small compared with the message, by the encrypted channel is a much simpler problem requiring less time than transmission of the message itself or of the key of the same length. This offers much promise for practical applications. Flexibility of block ciphers enables their usage to build other cryptographic primitives: pseudorandom sequence generator, stream cipher, message authentication code, cryptographic hashes.

In general, a block cipher consists of two algorithms forming a pair: encryption and decryption [2]. Both algorithms are represented as functions. At the input of the encryption function E arrives the data block M, with a size of n bits, and the key K, k bits in size. At the output we have the cipher text block C with a size of n bits.

For the key K, E_K is a bijective function for a set of k-bit blocks. After the encryption operation E, the code C and the key K arrive to the input of the decryption function D. At the output we have the data block M. The decryption function D is an inverse to the encryption

*E-mail: SidorenkoA@yandex.ru

function E

$$D = E^{-1}, \tag{1}$$

$$\forall\ K\ :\ D_K\left(E_K\left(M\right)\right) = M \tag{2}$$

$$\text{and } E_K\left(D_K\left(C\right)\right) = C. \tag{3}$$

Here for encryption and decryption the same key is used due to the symmetric block cipher.

3. Implementation of the encryption algorithm based on dynamic chaos

In this paper we propose an algorithm and a block encryption scheme based on discrete chaotic maps using the Feistel network structure. The PCBC encryption mode (Propagating Cipher Block) has been used [3]: block length is 32 bits; key length is 128 bits.

The encryption block diagram is shown in figure 1. By the proposed scheme, the encryption is realized in 14 rounds. At every round we use a subkey including four components: $K_i(1)$, $K_i(2)$, $K_i(3)$, $K_i(4)$. A length of each subkey is 4 bytes (1 byte for every subkey component). The subkey generation follows a particular algorithm. We suggest to use a combination of two chaotic maps iteratively invoked n times as the round function F (figure 2). The location of the permutation function to process 32 bits of information once at the beginning and at the end of the encryption procedure in the algorithm is defined separately.

Figure 2 shows the internal structure of the round function F, where

⊕ – XOR operation,
⊞ – modulo 28 summation operation,
f1 and **f2** are chaotic mappings.

In the proposed encryption algorithm, two chaotic mappings are used: two-dimensional cubic mapping (function **f1**) and logistic mapping (function **f2**).

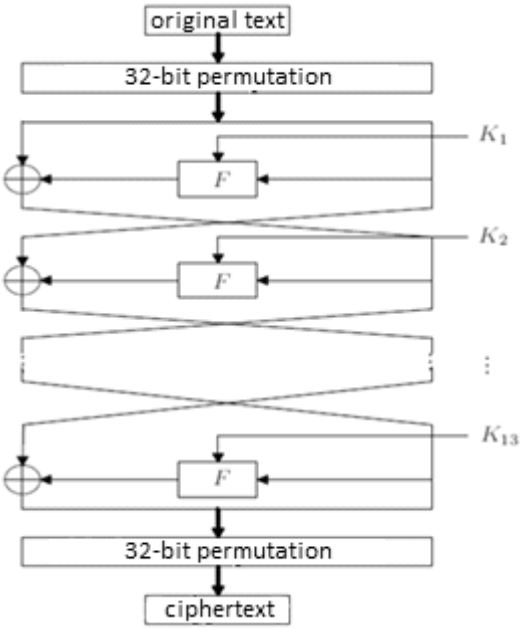

FIG. 1: Structural encryption scheme.

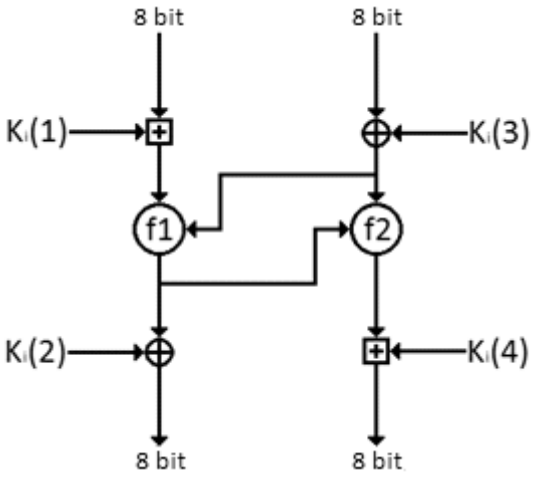

FIG. 2: Internal structure of the round function F.

A two-dimensional cubic mapping having some properties of the Duffing oscillator with a negative stiffness is mathematically described by the formulas

$$x_{n+1} = y_n, \tag{4}$$

$$y_{n+1} = -bx_n + dy_n - y_n^3. \tag{5}$$

For the parameters $b = 0.2$ and $d = 2.77$, the two-dimensional cubic map reveals chaotic properties.

Logistic mapping (also quadratic or Feigenbaum map) is a polynomial map with the mathematical expression of the form

$$x_{n+1} = rx_n(1 - x_n) \tag{6}$$

where x_n ranges from 0 to 1, r is the control parameter.

For the values ranging $3.57 < r < 4.0$, the behavior of the logistic map is chaotic.

We have implemented the proposed algorithm using the language C++. The results were obtained on the Intel Core i7-4700MQ 2.4 GHz with 8 GB RAM

4. Security analysis

The ability of a cryptosystem to withstand the attempts of an unauthorized access to gain the unencrypted information is the critical factor. A good cryptosystem should resist all kinds of known attacks, such as known/chosen plain text attack, cipher text only attack, statistical attack, differential attack, various brute-force attacks.

4.1. Histogram analysis

Histogram of an image shows the image pixels distribution at each level. The distribution of a cipher text is very important as it should hide the redundancy of the plain text and should not leak any information about the plain text or about the relationship between plain text and cipher text.

As the test images are using the black image (figure 3a) and the faculty building image (figure 4a)

From figures 3 and 4 it is seen that histograms of the encrypted images are independent of the original image type and differ greatly from their original histograms. Thus, the proposed algorithm provides no useful statistical data in the encrypted image and is resistant to various kinds of statistical attacks.

4.2. Correlation of adjacent pixels

For the ordinary image having the particular visual content, each pixel is highly correlated with its adjacent pixels in horizontal, vertical or diagonal direction. However, an efficient image cryptosystem should produce the encrypted image with a sufficiently low correlation between the adjacent pixels.

To quantify and compare correlations of the adjacent pixels in the plain and cipher images, the following procedure has been carried out. First, we randomly select 10,000 pairs of adjacent pixels in each direction in the plain image and in the corresponding ciphered image. Then, we calculate the correlation coefficient r_{xy} for each pair using the following formulas:

$$r_{xy} = \frac{|\mathrm{Cov}\,(x, y)|}{\sqrt{D\,(x)}\sqrt{D\,(y)}}, \tag{7}$$

$$E\,(x) = \frac{1}{N}\sum_{i=1}^{n} x_i, \tag{8}$$

$$D\,(x) = \frac{1}{N}\sum_{i=1}^{n} (x_i - E\,(x))^2, \tag{9}$$

$$\mathrm{Cov}\,(x, y) = \frac{1}{N}\sum_{i=1}^{n} (x_i - E\,(x))\,(y_i - E\,(y)) \tag{10}$$

where x and y represent the values of two neighboring R (red), G (green) or B (blue) components of the pixel.

Figure 5 shows the distribution for 10,000 randomly selected pairs of the adjacent pixel components in the original and in the encrypted Faculty-building images.

Graphic results indicate that there is no apparent connection between pixels of the encrypted image.

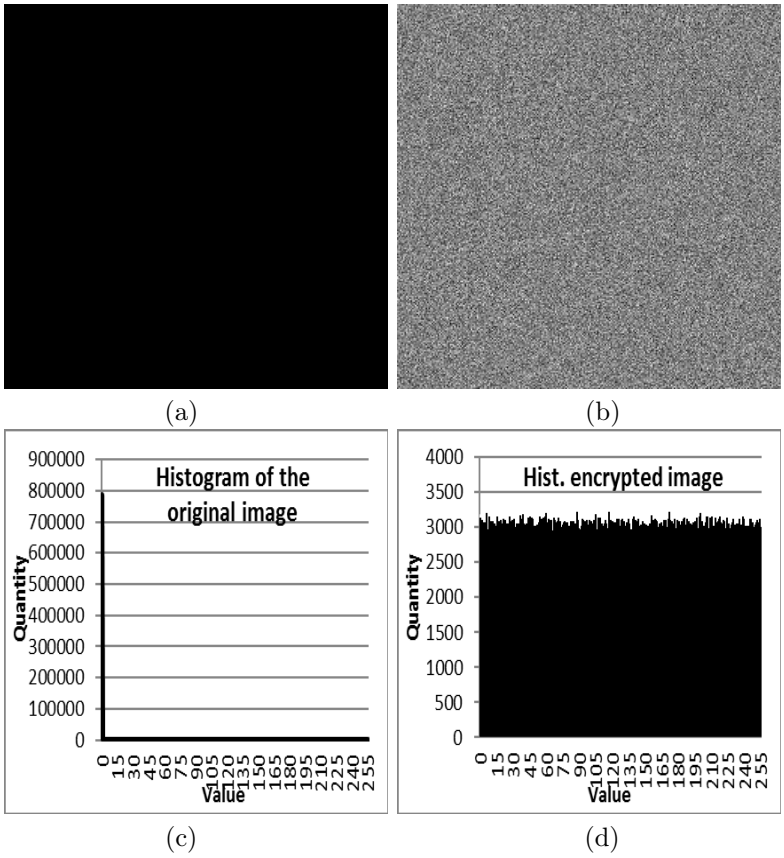

FIG. 3. Histogram analysis of the first image 512x512 pixels in size: (a) original image, (b) encrypted image, (c) histogram of the original image, (d) histogram of the encrypted image.

As one can see from Table 1, in the encrypted image the dependence of the neighboring pixels is much smaller than that in the original image.

4.3. Information entropy

In information theory, the entropy is the most significant feature of disorder or, more precisely, unpredictability. To calculate the entropy $H(S)$ of the source S, we use the formula

$$H(S) = -\sum_{i=1}^{N} p(s_i) \log_2 p(s_i) \qquad (11)$$

where N is the number of bits for representation of the source, and $p(s_i)$ represents the probability of the symbol s_i so that the entropy be expressed in bits.

For a truly random source emitting 2N symbols, the entropy is $H(S) = N$. Therefore, for a ciphered image with 256 gray levels, ideally the entropy should be $H(S) = 8$. If the entropy of the symbols generated at the output is below 8, there exists a certain degree of predictability threatening the information security.

Testing of the algorithm has shown that the entropy for the first image and for the second image was 0 and 6.44381, respectively. However, entropies of the encrypted first and second images were 7.99977 and 7.99976. The entropy of the encrypted images was very close to the ideal value of 8.

In this way the encryption algorithm is robust to statistical attacks. Information leakage in the encryption process is negligible.

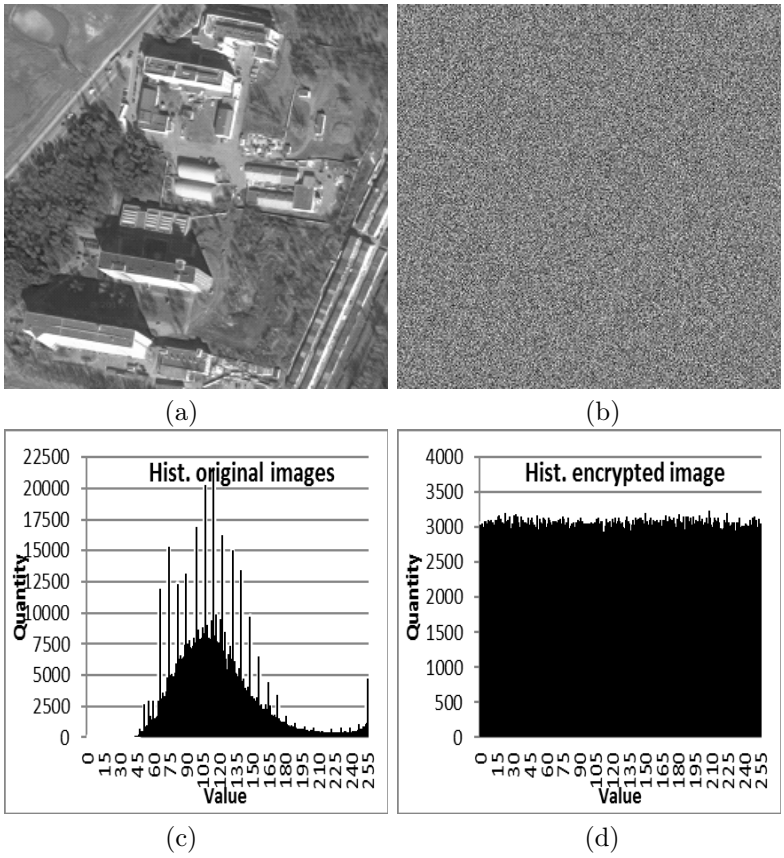

FIG. 4. Histogram analysis of the second image 512×512 pixels in size: (a) original image, (b) encrypted image, (c) histogram of the original image, (d) histogram of the encrypted image.

Table 1. Correlation coefficients of the original (the first one) and encrypted (second one) images with dimensions 512×512 pixels when using the proposed algorithm.

Direction	Faculty-building image					
	Original image			Encrypted image		
	R	G	B	R	G	B
Horizontal	0.961967	0.969685	0.964691	0.001886	0.001767	0.000172
Vertical	0.947319	0.957369	0.951228	0.002445	0.000473	0.001050
Diagonal 1	0.920903	0.935937	0.926406	0.001370	0.001668	0.000467
Diagonal 2	0.916809	0.932275	0.922862	0.000501	0.000756	0.001503

4.4. Avalanche effect

Sensitivity to the plain text is tested by encryption of a color image. First one bit is changed in the original image. Then the modified image is encrypted again using the same key. The change indicator for a couple of bits of the encrypted image is obtained as follows: [4] [5]:

$$\textit{Bit change indicator} =$$

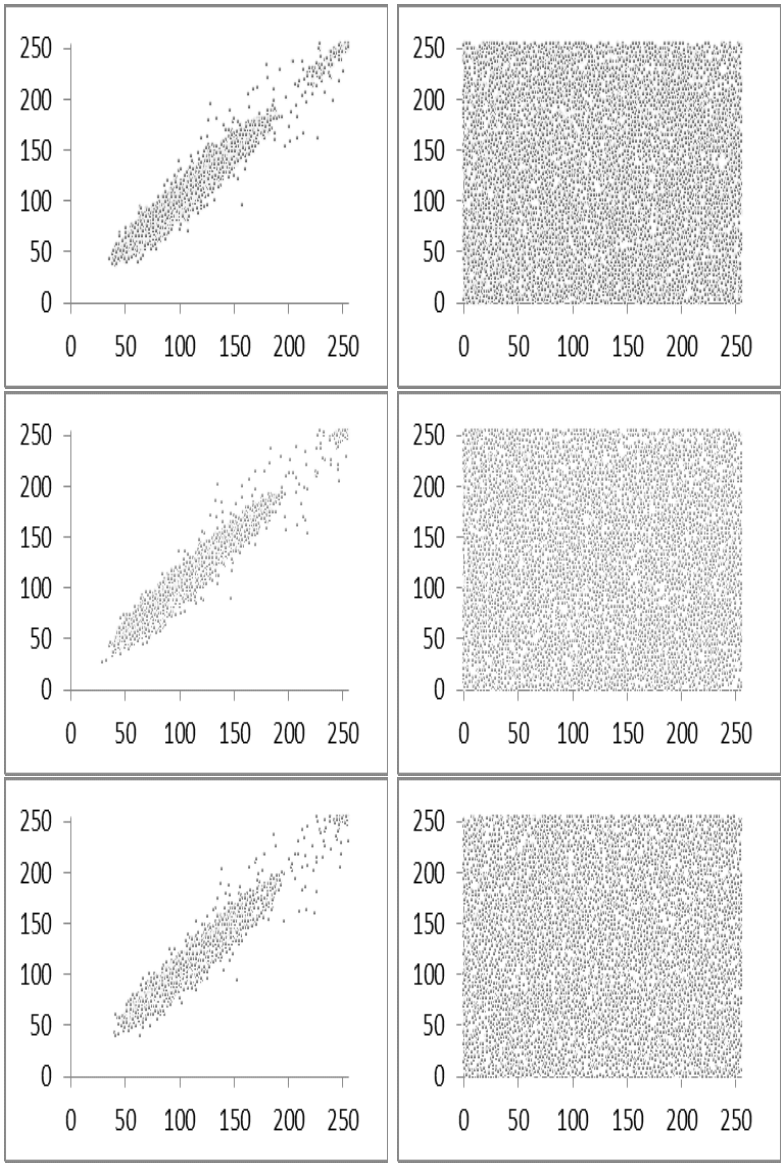

FIG. 5. Distribution of two horizontally adjacent pixels in the original second image: (a) red, (c) green, and (e) blue components. Distribution of two horizontally adjacent pixels in the encrypted second image: (b) red,(d) green, and (f) blue components.

$$= \frac{Number\ of\ bits\ changing\ value}{Total\ bit\ number}. \qquad (12)$$

As demonstrated by the results of the proposed algorithm testing, a change of one bit in the source image results in changing of 49.9978% of the bits in the encrypted image, that is very close to the ideal value of 50%. Thus, the algorithm is robust to linear attacks.

4.5. Resistance to differential attacks

Based on the principles of cryptology, a good encryption algorithm should be sensitive to the plain text. Sensitivity of the encryption algorithm can be quantified as the Number of Pixels Change Rate (NPCR), and the Unified Average Changing Intensity (UACI) [1]. Thereafter, NPCR and UACI may be determined by the following

formulas:

$$NPCR = \frac{1}{M\ N}\sum_{i=1}^{M}\sum_{j=1}^{N} D\left(i,j\right)100\%, \quad (12)$$

$$UACI = \frac{1}{255\ M\ N}$$

$$\times\sum_{i=1}^{M}\sum_{j=1}^{N}\left|C_1\left(i,j\right) - C_2\left(i,j\right)\right|100\%, \quad (13)$$

where $D\left(i,j\right) = \begin{cases} 0, \text{ if } C_1\left(i,j\right) = C_2\left(i,j\right) \\ 1, \text{ if } C_1\left(i,j\right) \neq C_2\left(i,j\right) \end{cases}$; $C_1\left(i,j\right)$ and $C_2\left(i,j\right)$ are pixels of two scrambled images at the position $(i,\ j)$. M and N represent the number of the image rows and columns, respectively.

The ideal values of NPCR and UACI can be calculated by the following formulas [6]:

$$NPCR_{ideal} = \left(1 - 2^{-n}\right)100\% \quad (14)$$

$$UACI_{ideal} = \frac{1}{2^{2n}}\frac{\sum_{i=1}^{2^n-1} i\left(i+1\right)}{2^n - 1}100\%$$

$$= \frac{1}{3}\left(1 + 2^{-n}\right)100\% \quad (15)$$

where n is a number of bits used to represent one pixel. In the gray image we use 8 bits per pixel, $n = 8$.

Tables 2 and 3 demonstrate the values of UACI and NPCR for two encrypted second images 512×512 pixels in size, provided that the original images differ in one pixel.

Based on Tables 2 and 3, it is seen that the resulting values of NPCR and UACI fluctuate around the ideal values. So, the algorithm is sensitive to small changes in the original image, being resistant to differential attacks.

Table 2. Calculation results for NPCR of two encrypted second images, provided the original images differ in one pixel.

Pixel changes	NPCR			
	R	G	B	Ideal
(1.1)	99.60	99.61	99.61	
(512.1)	99.58	99.59	99.61	
(1.512)	99.63	99.63	99.59	99.61
(512.512)	99.61	99.62	99.61	
(256.256)	99.58	99.59	99.60	

Table 3. Calculation results for UACI of two encrypted second images, provided the original images differ in one pixel.

Pixel changes	UACI			
	R	G	B	Ideal
(1.1)	33.39	33.46	33.41	
(512.1)	33.46	33.42	33.45	
(1.512)	33.46	33.38	33.46	33.46
(512.512)	33.52	33.45	33.50	
(256.256)	33.52	33.47	33.52	

5. Conclusion

As a result of this work, the data encryption algorithm based on dynamic chaos has been implemented. The features of this algorithm have been described in detail; the mappings used have been given. The effectiveness of this algorithm has been determined on the basis of quantitative parameters: information entropy, numerical characteristics of the byte distribution in the open and in the encrypted text; correlation; percentage of the value changing bits (avalanche effect); number of pixel change rate (NPCR) and unified average changing intensity (UACI).

It may be concluded that the encryption algorithm is resistant to various cryptographic attacks including statistical attack, differential attacks, brute force attacks, etc.

The claimed method allows to increase the degree of information protection, the encryption

efficiency and resistance to various cryptographic attacks. The implemented algorithm can be easily integrated into the hardware and can be used in a wide range of applications. The claimed method represents enormous functionality at the decision of cryptographic tasks.

References

[1] Xuanping Zhang, Xing Fan and JiayinWang. A chaos-based image encryption scheme using 2D rectangular transform and dependent substitution. Multimedia Tools and Applications. **75**, no.4, 1745-1763 (2016).

[2] Thomas Cusick and Pantelimon Stanica. Cryptographic Boolean functions and applications. Academic Press, 158–159 (2009).

[3] Bruce Schneier. Applied Cryptography. Protocols, algorithms, source code in C. Triumph, 146-161 (1994).

[4] Seyed Seyedzadeh and Yasaman Hashemi. Image Encryption Algorithm Based on Choquet Fuzzy Integral with Self-Adaptive Pseudo-Random Number Generator. 11th International Conference on Intelligent Systems Design and Applications (ISDA), 642 – 647 (2011), IEEE Xplore, DOI:10.1109/ISDA.2011.6121728

[5] Seyed Seyedzadeh, Benyamin Norouzi and Sattar Mirzakuchaki. RGB Color Image Encryption based on Choquet Fuzzy Integral. The Journal of Systems and Software. **97**, 128–139 (2014).

[6] Jianfeng Zhao et al. A novel image encryption scheme based on an improper fractional-order chaotic system. Nonlinear Dynamics. **80**, no. 4, 1721-1729 (2015).