Cryptography Using Quasi Group and Chaotic Maps

Eng. Heba A. Abughali[1*], Mohammed A. Mikki[2]

[1] Computer Department, Palestine Technical Collage, Deir Elbalah, Palestine
[2] Computer Engineering Department, Islamic University, Gaza, Palestine

Email Address

eng.hghali@gmail.com (Eng. Heba A. Abughali), mmikki@iugaza.edu.ps (Mohammed A. Mikki)
*Correspondence: eng.hghali@gmail.com

Received: 28 December 2017**; Accepted:** 15 January 2018**; Published:** 2 Feburary 2018

Abstract:

In this paper a symmetric key (stream cipher mode/ block cipher mode) cryptosystem is proposed, involving chaotic maps and quasi group. The proposed cryptosystem destroys any existing patterns in the input, and also, it maximizes entropy. Moreover, the n-grams illustrate that the proposed cryptosystem is secure against the statistics analysis. Furthermore, Experimental results show that the ciphertext has good diffusion and confusion properties with respect to the plaintext and the key, also the results demonstrate that the block cipher mode gives higher entropy than the steam cipher mode.

Keywords:

Cryptography, Chaotic Map, Quasi Group, Block Cipher, Stream Cipher, Entropy, N-Grams, Autocorrelation

1. Introduction

Security is one of fundamental importance in digital communication. Hence, Cryptography is one of the most important fields in computer security. It’s a process of transmission data through unsecured channels, and only the authenticated receiver who has the legitimate key can read the encrypted messages which might be documents, phone conversations, images or other form of data. In cryptosystems the information must be scrambled, so that other users will not be able to access the actual information. While providing privacy remains a central goal, the field has expanded to encompass many others, including not just other goals of communication security, such as guaranteeing integrity and authenticity of communications, but many more sophisticated and fascinating goals. Once largely the domain of the military, cryptography is now in widespread use, and you are likely to have used it even if you don’t know it. When you shop on the Internet, for example to buy a book, cryptography is used to ensure privacy of your credit card number as it travels from you to the shop’s server. Or, in electronic banking, cryptography is used to ensure that your checks cannot be forged.

A chaotic map is a map that exhibits some sort of chaotic behavior. Maps may be parameterized by a discrete-time or a continuous-time parameter. Discrete maps usually take the form of iterated functions. Their properties, such as sensitive dependency on initial conditions and system parameters, and random-like outputs, are similar to confusion and diffusion cryptography properties, so they have been used to build good cryptosystems. Furthermore, these properties make the chaotic cryptosystems robust against any statistical attack. [1-3]

There are some similar properties between chaotic systems and traditional cryptographic algorithms. In cryptographic algorithms, diffusion and confusion are applied on plaintext over the encryption rounds of the algorithm. In chaotic systems, similar things happen on the initial input parameters. After a sufficiently large number of iterations, an input parameter will be eventually spread over the entire phase space through the random-like orbit over iterations. The stochasticity property of a chaotic system is similar to the diffusion and confusion properties of cryptographic algorithms.

For cryptographic algorithms, in order to decrypt the cipher text to the original plain text correctly, the same key should be used in both encryption and decryption. This is just similar to the requirement that chaotic systems need the same input parameter to reproduce the same output orbit. In this case, the system parameters and initial conditions can be considered as the private key of a chaotic cryptosystem. The table 1 summaries the common relationship which promotes chaos theory into practical cryptographic design. [4, 5].

Table 1. *A comparison of some features characterized by chaotic system and traditional cryptosystems.*

Chaotic system	**Traditional cryptosystems**
Ergodicity	Confusion
Sensitivity to initial condition and system parameters	Diffusion
Parameters	Encryption key
iterations	Cipher rounds

Quasi groups (or Latin squares) provide a powerful technique for generating a larger set of permutation transformations by permuting not only the samples but also transforming the amplitudes themselves across their range [6]. By doing this, they provide an immensely large number of keys, even for small alphabets. Therefore, quasi group based ciphers can have a variety of applications, and in some cases can be competitive to number theory based systems in terms of the difficulty they offer to brute force attacks. Moreover, it is resist to the statistical attack. [7,8]

Many researches in the field of using chaotic maps in cryptography have been developed. A cryptosystem with a private key block cipher algorithm used an external key of variable length (maximum 128-bits) to generate the system parameters and the initial conditions of the chaotic map were proposed. The ciphertext depends on the private key only [9]. The main weaknesses of this technique are the ciphertext depends on the private key only, so it is vulnerable to known plaintext attack. It uses small block size (8 bits), and easy to brute force because the initial condition only 256 values.

Later, [10] explained the weaknesses of using a 128-bit external key to derive the initial conditions and number of iterations. These weaknesses are summarized as follows: The interval chosen for a system parameter (λ), the small resolution used to calculate it, together with the deterministic nature of the algorithm, allow for a known plaintext attack. Also, the process to derive an initial condition (X) and number of iterations (N) from the external key (K) is fundamentally flawed, allowing for chosen ciphertext and chosen plaintext attacks. It is concluded from these facts that the total lack of security along with the low encryption speed discourages the use of this algorithm for secure applications.

In 2005, another chaotic map based technique was proposed, in which multiple one-dimensional chaotic maps are used instead of a one-dimensional chaotic map. This algorithm uses an external secret key of variable length (maximum 128-bits). The plaintext is divided into groups of variable length (number of blocks in each group is different). These groups are encrypted by using randomly chosen chaotic map from a set of chaotic maps. The number of iterations and initial conditions for the chaotic maps depend on the randomly chosen session key and on the previous block of ciphertext. The encryption/decryption process is governed by two dynamic tables, which contain the number of iterations and initial conditions for the chaotic maps, these tables are updated from time to time during the encryption/decryption process [11]. In this cryptosystem, the ciphertext depends on the private key only, so it is vulnerable to known plaintext attack and also a small block size (8 bits) is used.

Although the previous algorithm is good regarding the confusion and diffusion as well as efficiency, in 2007, a cryptanalysis technique showed some weaknesses of it, which can be summarized as follows: The cipher generated looks like a block cipher, but it behaves as a stream cipher ,and equations of initial conditions and number of iterations are dependent on the secret key only, which results in the initial contents of the two dynamic tables exactly the same for different plaintext sequences as long as the secret key is fixed. The variable, the initial condition, which changes by iterating the maps is used to update the two tables for encrypting the next plaintext block, and each plaintext block is encrypted with the last value of X . At the end of the paper, they make a straightforward modification to make the value of X dependent on both the key and the plain text [12].

A private key cryptosystem is publicized in [13], an improved cryptosystem has been proposed to show the essential weaknesses and redundancies of the previous chaotic cryptosystems. An improved scheme is used to eliminate these weaknesses by different approaches. This scheme uses two skew tent maps instead of a logistic map. The redundant operations in previous systems are abandoned to simplify the cipher. Permutation within ciphertext was implemented by using two independent chaotic variables to mask the plaintext. Here, in the improved chaotic scheme, a one-dimensional chaotic map is used, which limits the degree of confusion and diffusion. Also, the number of iterations is calculated using the key indices (60 to 67), i.e. K_{60}, K_{61},…., K_{67} stream is used to provide the number of iterations to be applied to the map, which reduces the range of the expected number of iterations (Maximum number of iterations = 2^8 (256) values).

On the other hand, quasi group is a non-associative group, has good scrambling properties. The approaches [14 - 16] using a quasi-group multiplication operation as a permutations. However, these schemes need another reliable encryption algorithm is required to preserve the secrecy of the encryption. It is necessary to transmit the quasi group that is being used for encryption, which is one of the main drawbacks of the above approach. Once the eavesdropper breaks the encapsulating cipher he has access to the quasi group used for the encryption and all the other required information to get the data.

Authors concluded that cryptosystem based on quasigroups are slowly but surely. Furthermore, they noticed that there are many broken designs based on quasigroups, but also there are some with perfect crypto properties [17, 18].

The authors realized a Laplace transformation based synchronization between two fractional-order chaotic systems to execute error-free encryption and decryption of digital images. The statistical analyses show the consistent encryption strength of the proposed algorithm in [19]. However, a careful probe of their algorithm uncovers underlying security shortcomings which make it vulnerable to cryptanalysis. In [20] chosen plaintext-attack/known plaintext-attack was proposed to break the algorithm completely. It is shown that the plain-image can be successfully recovered without knowing secret key.

We investigate the usage of chaotic maps in cryptography, their properties, such as sensitive dependency on initial conditions and system parameters, and random-like outputs, are similar to confusion and diffusion cryptography properties. Furthermore, quasi groups provide a powerful technique for generating a larger set of permutation transformations, so, we adopt the chaotic maps and quasi groups in our proposed cryptosystem.

2. Materials and Methods

The proposed private key cryptosystem is based on chaotic maps and quasi groups, it has two types: stream cipher mode and block cipher mode, in the following sections those modes and their security analysis will be explored.

2.1. The Proposed Private Key Cryptosystem (Stream Cipher Mode)

Here, we use both of chaotic maps and quasi group to build an efficient cryptosystem.

The general block diagram, illustrates the phases of the cryptosystem is shown in Fig. 1. The encryption process consists of five stages, the first phase is initialization, where the variables, chaotic maps and quasi groups are built, then preprocessing phase where the message is converted into numbers, while in chaotic transposition phase, the logistic map is used to rearrange the message, the Quasi transposition phase mix up the massage using quasi group, the last phase is substitution in which the chaotic maps are used in order to change the cipher values to be unreadable date.

The steps involved in the proposed encryption/decryption process are given below.

Initialization Stage:

1. Getting the secret key

2. Build the quasi groups in the encryption process, and build the inverse quasi group in the decryption process.

3. Generate variables using chaotic maps

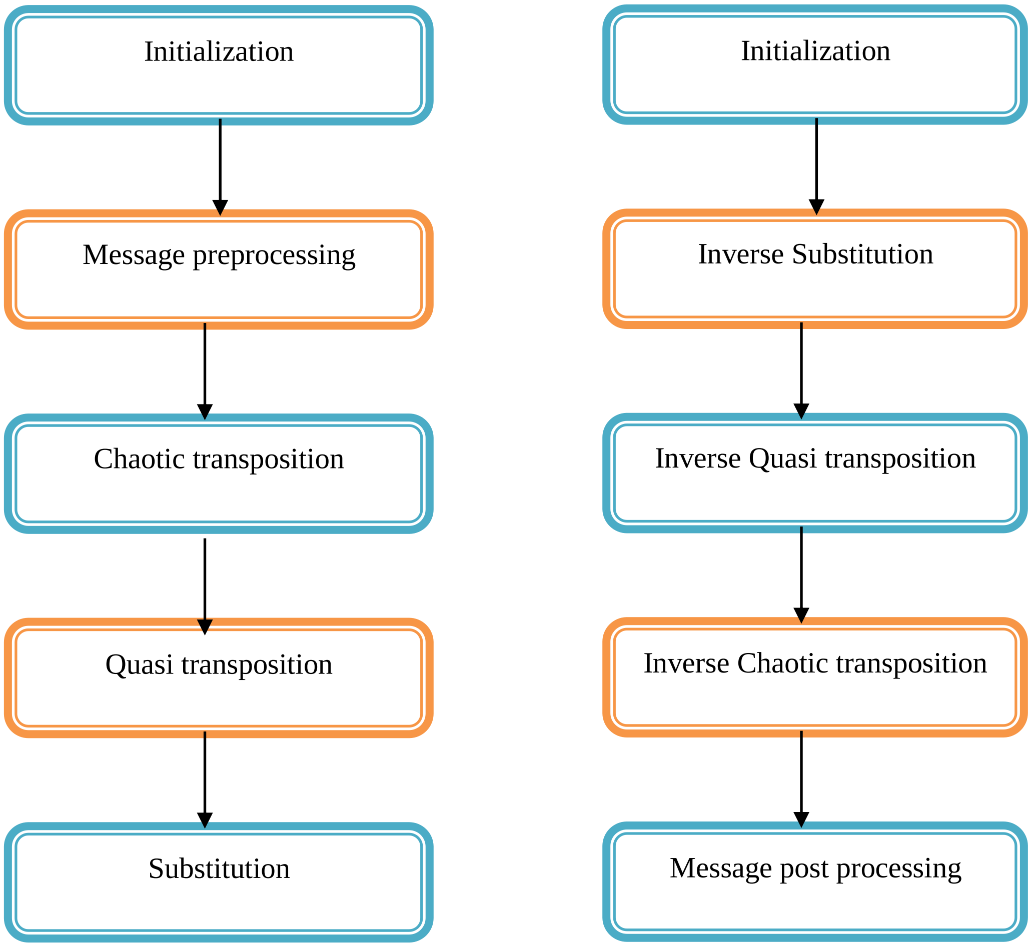


Figure 1(a). *Encryption*

Figure 1(b). *Decryption*

Figure 1. *(a) Block diagram showing the private key cryptosystem encryption part, (b) block diagram showing the private key cryptosystem decryption part*

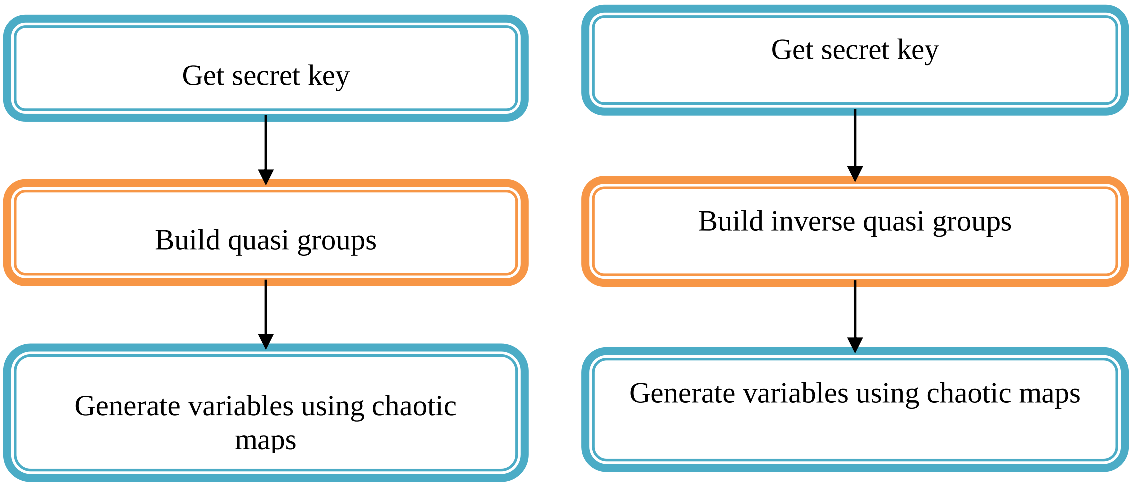

Get secret key
Build quasi groups
Generate variables using chaotic maps
Get secret key
Build inverse quasi groups
Generate variables using chaotic maps

Figure 2.*(a) Encryption*

Figure 2.*(b) Decryption*

Figure 2. *(a) Block diagram showing the initialization steps in encryption part, (b) Block diagram showing the initialization steps in decryption part*

<u>Encryption process:</u>

Purpose: encrypting the data.

Inputs: the original message.

Outputs: the encrypted message.

Procedure:

Step 1: Initialize the required variables.

Step 2: Convert the message into numbers.

Step 3: Mix up the plaintext using chaotic maps transposition.

Step 4: Shuffle the plaintext via quasi groups transposition.

Step 5: Apply chaotic maps substitution on the plaintext.

Decryption process:

Purpose: decrypting the message.

Inputs: the encrypted message.

Outputs: the original message.

Procedure:

Step 1: Initialize the required variables.

Step 2: Apply inverse chaotic maps substitution on the ciphertext.

Step 3: Apply inverse quasi groups' transposition.

Step 4: Apply inverse chaotic map transposition.

Step 5: Message post processing.

The initialization steps in encryption side are demonstrated in Fig. 2.a, while Fig. 2.b shows the initialization steps in decryption side.

The different between them is the second step in the encryption side the construction of quasi groups, but in the decryption side the construction of inverse quasi groups.

2.2. The Proposed Private Key Cryptosystem (Block Cipher Mode)

Here we examine the proposed cryptosystem in block cipher mode.

It's known that a stream cipher mode is extremely simple and fast, but requires synchronization, while a block cipher mode output is random-looking and has good statistical properties.

The message is divided into 128 byte blocks and each block is encrypted separately using particular key, the user determines the number of encryption boxes, in the following experiments I have four encryption boxes with different keys as shown in Fig. 3.

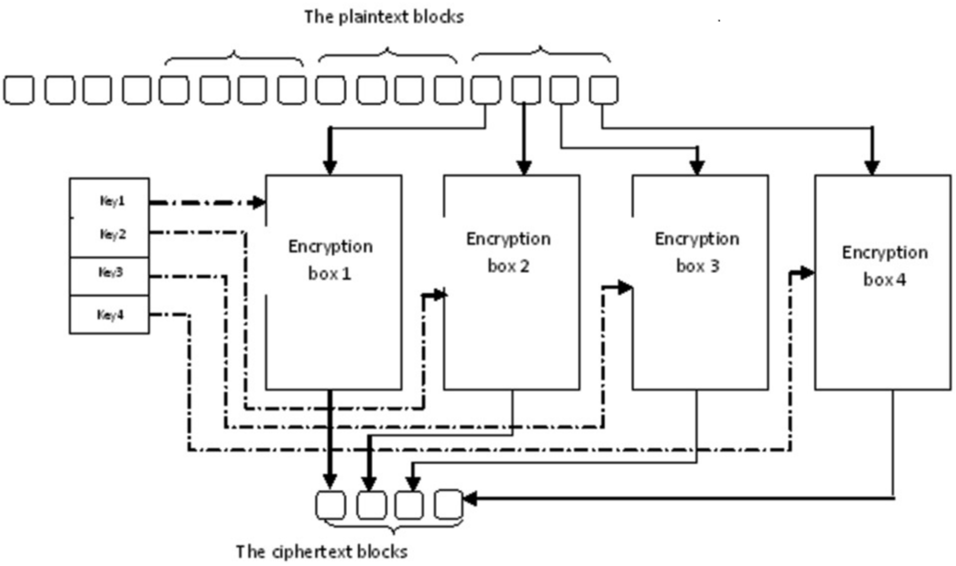


Figure 3. *The proposed private key cryptosystem in block cipher mode*

Each encryption box works individually with different key, so more confusion and diffusion are obtained.

3. Results and Discussion

The proposed cryptosystem has been implemented using Matlab and the simulation results were observed on Core2 Duo, 2.40GHz with 2 GB RAM , We have acquired outstanding results, as shown in the following figures the ciphertext and the plaintext are totally different.

3.1. The Proposed Private Key Cryptosystem (Stream Cipher Mode)

To show the performance of the proposed cryptosystem, we compared between two plain texts, the first one is consist of constant value particularly the letter 'E' and the other is ordinary text, Fig. 4 shows the first plaintext, and its ciphertext shown in Fig. 5, the second plain text shown in Fig. 6 while the resulted ciphertext shown in Fig. 7.

As observed from these figures, the ciphertext shows a complete random behavior.

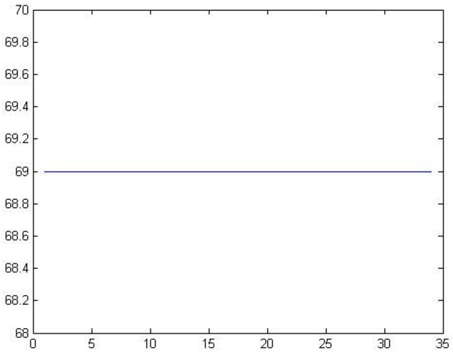


Figure 4. *The constant plaintext.*

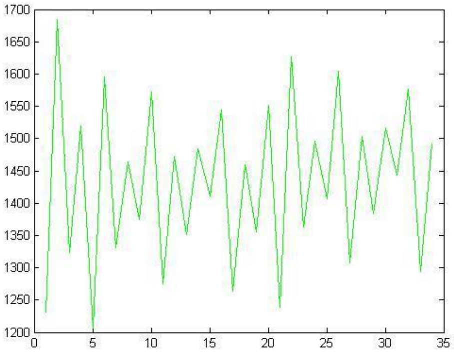


Figure 5. *The ciphertext of constant plaintext.*

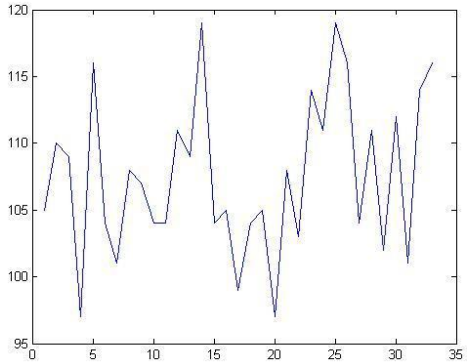


Figure 6. The case 2 plaintext

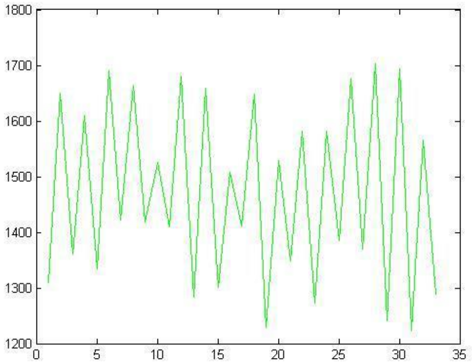


Figure 7. The ciphertext of case 2

3.2. Security analysis (Stream Cipher Mode)

The following results demonstrate how much the entropy is maximized, and how the proposed cryptosystem does destroy any existing patterns in the input plaintext, which is desirable for a good cryptosystem.

3.2.1. Entropy

Entropy is a statistical measure of randomness that can be used to characterize the texture of the input. Entropy is a measure of disorder, or more precisely unpredictability.

English text has fairly low entropy. In other words, it is fairly predictable. Even if we don't know exactly what is going to come next, we can be fairly certain that, for example, there will be many more e's than z's, or that the combination 'qu' will be much more common than any other combination with a 'q' in it and the combination 'th' will be more common than any of them. [21]

Case 1:

The plaintext:

"inmathelkhhomwhichialgrowthofpertuxcvrbationslijintheirfyhvnitialconditionstheb ehiaaviityudjkflaorofthat"

The entropy of the plaintext = 3.7600, while the entropy of the ciphertext = 6.5466.

Case 2:

The plaintext:

"EEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE EEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE"

The entropy of the plaintext = 0, while the entropy of the ciphertext = 4.8580.

It's observed that the entropy is increased significantly, so the proposed cryptosystem destroys the characteristic of the sent message by increasing the randomness in it.

3.2.2. N-grams

N-grams model is a type of probabilistic model for predicting the next item in such a sequence. The N-gram analysis determines the frequency of different N-grams in a text. Especially the gaps between equal N-grams can potentially be very useful for cracking a cipher because they can point to the key length. [22]

An n-gram of size 2 is referred to as a "bigram" (or, less commonly, a "digram"); size 3 is a "trigram".[23]

Case 1:

The plaintext:

"inmathelkhhomwhichialgrowthofpertuxcvrbationslijintheirfyhvnitialcondition sthebehiaaviityudjkflaorofthat"

Table 2. *The histogram, bigram and trigram of the case 1 plaintext*

Nr.	**Histogram**			**Bigram**			**Trigram**		
1.	**I**	14	13.46%	**TH**	5	4.81%	**THE**	3	2.88%
2.	**H**	11	10.58%	**TI**	3	2.88%	**IAL**	2	1.92%
3.	**T**	11	10.58%	**IT**	3	2.88%	**TIO**	2	1.92%
4.	**A**	8	7.69%	**HI**	3	2.88%	**ION**	2	1.92%
5.	**O**	8	7.69%	**ON**	3	2.88%	**ITI**	2	1.92%
6.	**N**	6	5.77%	**IA**	3	2.88%	**HIA**	2	1.92%
7.	**R**	5	4.81%	**HE**	3	2.88%	**ONS**	2	1.92%
8.	**L**	5	4.81%	**AT**	3	2.88%	**ORO**	1	0.96%
9.	**E**	5	4.81%	**RO**	2	1.92%	**NTH**	1	0.96%
10.	**F**	4	3.85%	**HO**	2	1.92%	**OFP**	1	0.96%
11.	**V**	3	2.88%	**IN**	2	1.92%	**NST**	1	0.96%
12.	**C**	3	2.88%	**IO**	2	1.92%	**OFT**	1	0.96%
13.	**S**	2	1.92%	**OF**	2	1.92%	**OND**	1	0.96%
14.	**B**	2	1.92%	**AL**	2	1.92%	**OMW**	1	0.96%
15.	**U**	2	1.92%	**NS**	2	1.92%	**NSL**	1	0.96%
16.	**W**	2	1.92%	**MW**	1	0.96%	**LKH**	1	0.96%
17.	**D**	2	1.92%	**OW**	1	0.96%	**LIJ**	1	0.96%
18.	**J**	2	1.92%	**MA**	1	0.96%	**LGR**	1	0.96%
19.	**K**	2	1.92%	**ND**	1	0.96%	**LCO**	1	0.96%
20.	**M**	2	1.92%	**OR**	1	0.96%	**MAT**	1	0.96%
21.	**Y**	2	1.92%	**OM**	1	0.96%	**MWH**	1	0.96%
22.	**X**	1	0.96%	**NM**	1	0.96%	**NMA**	1	0.96%
23.	**P**	1	0.96%	**NI**	1	0.96%	**NIT**	1	0.96%
24.	**G**	1	0.96%	**NT**	1	0.96%	**NDI**	1	0.96%
25.				**RF**	1	0.96%	**OWT**	1	0.96%
26.				**VR**	1	0.96%	**RBA**	1	0.96%
27.				**VN**	1	0.96%	**VNI**	1	0.96%
28.				**VI**	1	0.96%	**VII**	1	0.96%
29.				**WH**	1	0.96%	**UXC**	1	0.96%
30.				**WT**	1	0.96%	**UDJ**	1	0.96%

The ciphertext:

"s@Hfù- --K±¾h8?© àcv5N`o$¼w6v>H ð%`6 ųo½5¾M Żd ûn ǫa-2[a₵ ·à1Ha+"

Table 3. *The histogram, bigram and trigram of the case 1 ciphertext*

Nr.	**Histogram**			**Bigram**			**Trigram**		
1.	**H**	4	19.05%	**NO**	2	9.52%	**SHF**	1	4.76%
2.	**O**	3	14.29%	**SH**	1	4.76%	**OWV**	1	4.76%
3.	**N**	2	9.52%	**OW**	1	4.76%	**OMZ**	1	4.76%
4.	**V**	2	9.52%	**OM**	1	4.76%	**UOM**	1	4.76%

5.	**Z**	1	4.76%	**UO**	1	4.76%	**VHU**	1	4.76%
6.	**S**	1	4.76%	**VH**	1	4.76%	**ZDN**	1	4.76%
7.	**U**	1	4.76%	**ZD**	1	4.76%	**WVH**	1	4.76%
8.	**W**	1	4.76%	**WV**	1	4.76%	**VNO**	1	4.76%
9.	**M**	1	4.76%	**VN**	1	4.76%	**OHA**	1	4.76%
10.	**C**	1	4.76%	**OH**	1	4.76%	**NOW**	1	4.76%
11.	**D**	1	4.76%	**CV**	1	4.76%	**HCV**	1	4.76%
12.	**F**	1	4.76%	**HC**	1	4.76%	**FKH**	1	4.76%
13.	**K**	1	4.76%	**HA**	1	4.76%	**DNO**	1	4.76%
14.	**A**	1	4.76%	**FK**	1	4.76%	**HFK**	1	4.76%
15.				**HF**	1	4.76%	**HUO**	1	4.76%
16.				**HU**	1	4.76%	**NOH**	1	4.76%
17.				**MZ**	1	4.76%	**MZD**	1	4.76%
18.				**KH**	1	4.76%	**KHC**	1	4.76%
19.				**DN**	1	4.76%	**CVN**	1	4.76%

It observed from the previous tables (table 2 and table 3), that the characteristics of the plaintext and ciphertext are totally different; mainly the occurrence of the characters is changed in the ciphertext from it in the plaintext.

Case 2:

The plaintext:

"EEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE
EEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE"

Table 4. *The histogram, bigram and trigram of the case 2 plaintext*

Nr.	**Histogram**			**Bigram**			**Trigram**		
1.	**E**	29	100%	**EE**	14	48.28%	**EEE**	9	31.03%

The ciphertext:

"+5aQS¯«N)©x¨((3¤F"

Table 5. *The histogram, bigram and trigram of the case 2 ciphertext*

Nr.	**Histogram**			**Bigram**			**Trigram**		
1.	**X**	1	20%	**XF**	1	20%	**SNX**	1	20%
2.	**S**	1	20%	**SN**	1	20%	**QSN**	1	20%
3.	**Q**	1	20%	**QS**	1	20%	**NXF**	1	20%
4.	**N**	1	20%	**NX**	1	20%			
5.	**F**	1	20%						

It is noticed as shown in table 5 that the letter 'E' didn't appear in the ciphertext and there is no character as much as a recurring character 'E' in the plaintext. So, the proposed cryptosystem is secure against the statistics analysis due to the results demonstrated in table 4 and table 5.

3.2.3. Autocorrelation

Autocorrelation is a mathematical tool for finding repeating patterns. such as the presence of a periodic signal. It refers to the correlation of a time series with its own past and future values. [24]

It may be observed from Figs. 8 and 10 that the output characteristics of both cases are a very similar even though the inputs are very different.

In the other side Fig. 9 demonstrates the pattern in case 2 text, while Fig. 10 shows how does the proposed cryptosystem destroy the existing pattern, which is desirable for a good cryptosystem.

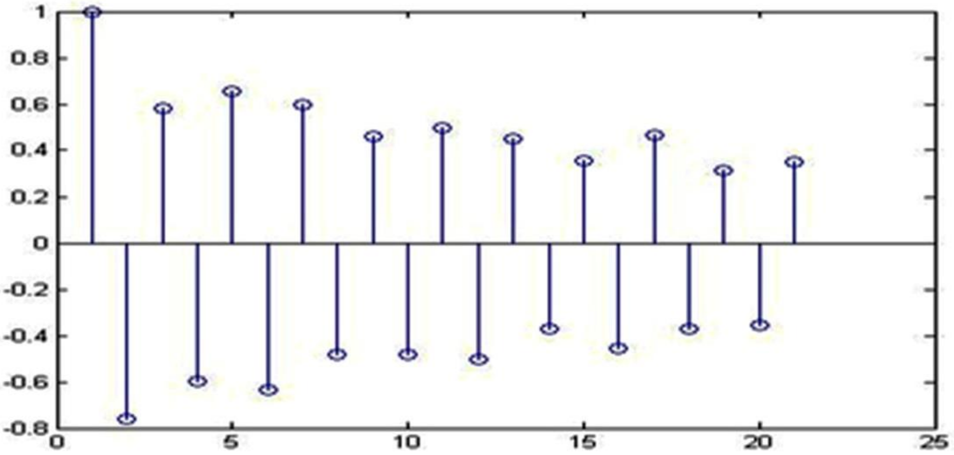


Figure 8. *The autocorrelation of case 1 ciphertext*

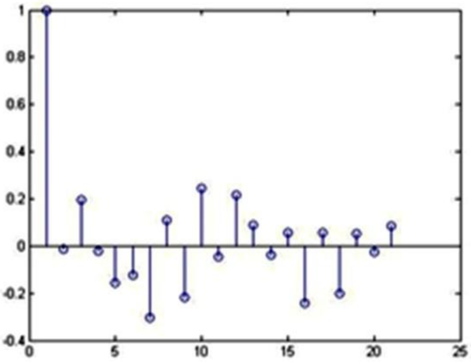


Figure 9. *The autocorrelation of case 2 plaintext*

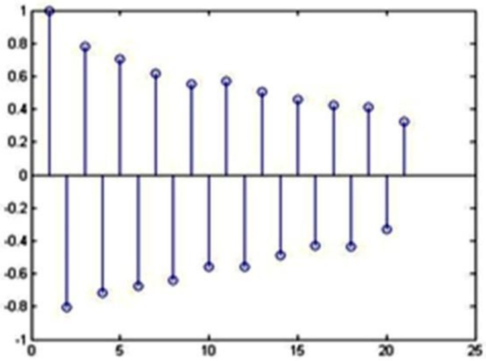


Figure 10. *The autocorrelation of case 2 ciphertext*

3.3. The Proposed Private Key Cryptosystem (Block Cipher Mode)

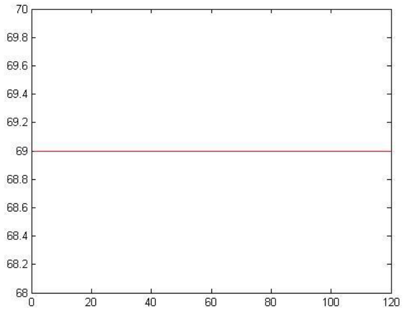
The experiments are applied in two plaintexts:

Case 1:

Plaintext: "EEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE"

Ciphertext:

"]eêyµ!}±Mƒ¤1§nû0êQ<è=× ¸IT¯pŒG« {c K3·"



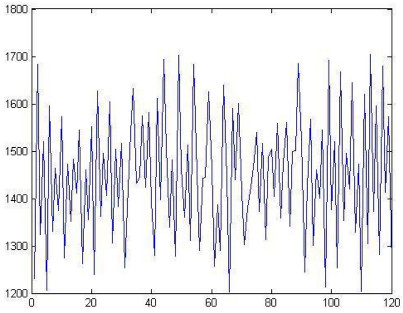


Figure 11. *Case 1 plaintext*

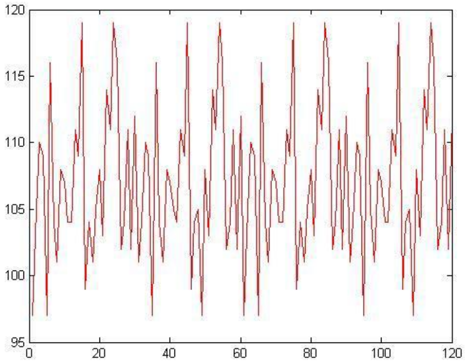
Figure 12. *Case 1 ciphertext*

Case 2:

Plaintext: "inmathelkhhomwchialgrowtfhofpeinmathelkhhomwchialgrowtfhofpein mathelkhhomwchialgrowtfhofpeinmathelkhhomwchialgrowtfhofpe "

Ciphertext:

" ?{[®d I5 g YQ ' ïNü·¾× ƒµ ¨çWü ¾k₵ =-"



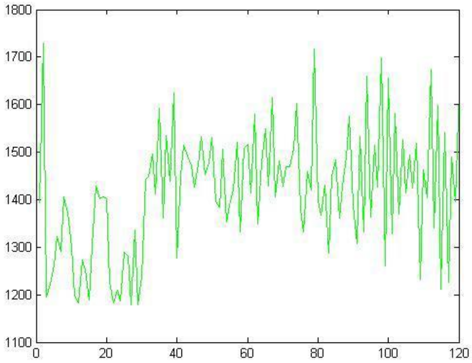


Figure 13. *Case 2 plaintext*

Figure 14. *Case 2 ciphertext*

It may be observed from the figures, that the encrypted sequence is essentially one value, even though the output sequence is very random. Study the characteristics of the output data in the next section.

3.4. Security analysis (Block cipher mode)

The following results demonstrate how much the entropy is maximized, and how the proposed cryptosystem does destroy any existing patterns in the input plaintext, which is desirable for a good cryptosystem.

3.4.1. Entropy

Case 1:

The entropy of the plaintext = 0, while the entropy of the ciphertext = 6.6736.

Case 2:

The entropy of plaintext= 3.7600, while the entropy of the ciphertext= 6.6506.

3.4.2. N-grams

It's clear that the cryptosystem destroy any patterns , for example in plaintext case 1 the letter "E" has histogram 100% as shown in table 6 , the histogram reduced to 9.09 % in the ciphertext witch clear in table 7.

The same results get in case 2 the letters (H , O, A, L and M)which are in the plaintext disappeared in the ciphertext, while the letter "W" has histogram 6.67% in the plain text as shown in table 8 , became 12.5% in the ciphertext as mentioned in table 9.

Case 1:

Table 6. *The histogram, bigram and trigram of case 1 plaintext*

Nr.	**Histogram**			**Bigram**			**Trigram**		
1.	**E**	120	100%	**EE**	58	50 %	**EEE**	9	32.74%

Table 7. *The histogram, bigram and trigram of case 1 ciphertext*

Nr.	**Histogram**			**Bigram**			**Trigram**		
1.	**P**	1	9.09%	**PG**	1	9.09%	**QIT**	1	9.09%
2.	**Q**	1	9.09%	**QI**	1	9.09%	**TPG**	1	9.09%
3.	**T**	1	9.09%	**TP**	1	9.09%	**YMN**	1	9.09%
4.	**Y**	1	9.09%	**YM**	1	9.09%	**PGC**	1	9.09%
5.	**N**	1	9.09%	**NQ**	1	9.09%	**NQI**	1	9.09%
6.	**M**	1	9.09%	**MN**	1	9.09%	**GCK**	1	9.09%
7.	**E**	1	9.09%	**EY**	1	9.09%	**ITP**	1	9.09%
8.	**G**	1	9.09%	**GC**	1	9.09%	**MNQ**	1	9.09%
9.	**I**	1	9.09%	**IT**	1	9.09%	**EYM**	1	9.09%
10.	**K**	1	9.09%	**CK**	1	9.09%			
11.	**C**	1	9.09%						

Case 2:

Table 8. *The histogram, bigram and trigram of case 2 plaintext*

Nr.	**Histogram**			**Bigram**			**Trigram**		
1.	**H**	20	16.67%	**HO**	8	6.67%	**NMA**	4	3.33%
2.	**O**	12	10%	**NM**	4	3.33%	**OFP**	4	3.33%
3.	**A**	8	6.67%	**OF**	4	3.33%	**MWC**	4	3.33%
4.	**L**	8	6.67%	**MW**	4	3.33%	**MAT**	4	3.33%
5.	**M**	8	6.67%	**LK**	4	3.33%	**LGR**	4	3.33%
6.	**W**	8	6.67%	**MA**	4	3.33%	**LKH**	4	3.33%
7.	**I**	8	6.67%	**OM**	4	3.33%	**OMW**	4	3.33%
8.	**E**	8	6.67%	**OW**	4	3.33%	**OWT**	4	3.33%
9.	**F**	8	6.67%	**WC**	4	3.33%	**WCH**	4	3.33%
10.	**T**	8	6.67%	**WT**	4	3.33%	**WTF**	4	3.33%
11.	**R**	4	3.33%	**TH**	4	3.33%	**THE**	4	3.33%
12.	**P**	4	3.33%	**TF**	4	3.33%	**TFH**	4	3.33%
13.	**K**	4	3.33%	**PE**	4	3.33%	**ROW**	4	3.33%
14.	**C**	4	3.33%	**RO**	4	3.33%	**KHH**	4	3.33%
15.	**G**	4	3.33%	**LG**	4	3.33%	**INM**	4	3.33%
16.	**N**	4	3.33%	**KH**	4	3.33%	**FHO**	4	3.33%
17.				**FH**	4	3.33%	**FPE**	4	3.33%
18.				**FP**	4	3.33%	**ELK**	4	3.33%
19.				**EL**	4	3.33%	**CHI**	4	3.33%
20.				**CH**	4	3.33%	**ATH**	4	3.33%

Table 9. *The histogram, bigram and trigram of case 2 ciphertext*

Nr.	Histogram				Bigram				Trigram			
1.	**Q**	1		12.5%	**WK**		1		12.5%	**QNW**		1		12.5%
2.	**W**	1		12.5%	**YQ**		1		12.5%	**YQN**		1		12.5%
3.	**Y**	1		12.5%	**QN**		1		12.5%	**NWK**		1		12.5%
4.	**N**	1		12.5%	**NW**		1		12.5%	**IGY**		1		12.5%
5.	**K**	1		12.5%	**GY**		1		12.5%	**GYQ**		1		12.5%
6.	**G**	1		12.5%	**IG**		1		12.5%	**DIG**		1		12.5%
7.	**I**	1		12.5%	**DI**		1		12.5%					
8.	**D**	1		12.5%										

3.4.3. Autocorrelation

It may be noticed from Figures 15 and 17 that the output characteristics of both cases are a very similar even though the inputs are very different. In the other side Fig. 16 demonstrates the pattern in case 2 text, while Fig. 17 shows how does the proposed cryptosystem destroy the existing pattern, which is desirable for a good cryptosystem.

The result shows more randomness and confirms the diffusion property with block cipher more than it when using the stream cipher mode; on the other hand stream ciphers advantage is that they are much faster than block cipher.

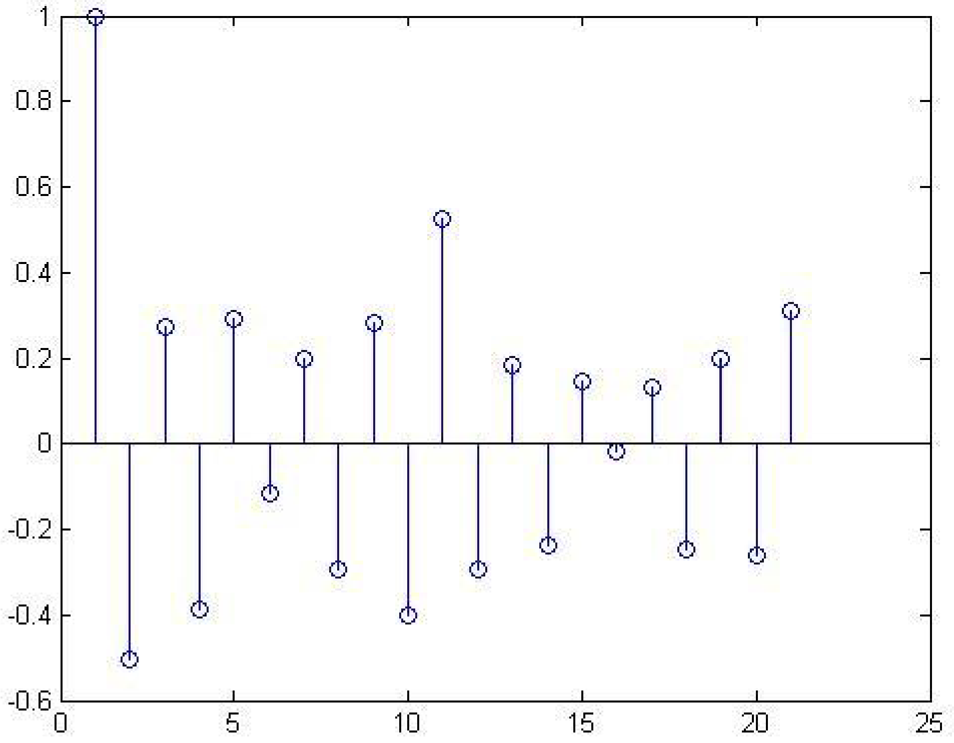
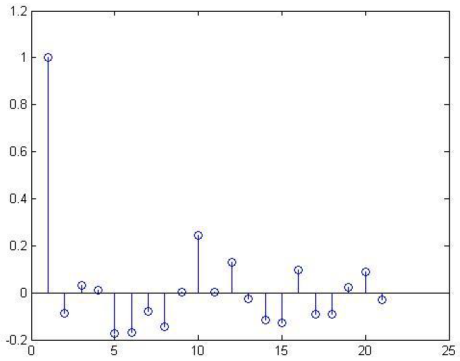


Figure 15. *Autocorrelation of case 1 ciphertext*



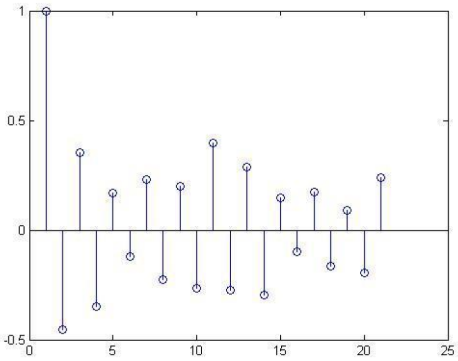


Figure 16. *Autocorrelation of case 2 plaintext.*

Figure 17. *Autocorrelation of case 2 ciphertext.*

4. Conclusions

In this paper proposed private key cryptosystem in two modes, stream cipher and block cipher mode.

The encryption process is divided into five phases, first the parameters and quasi groups were defined in the initialization phase, and second the preprocessing phase was in the preparing phase, where the plaintext was converted into numbers, which considered more suitable to deal with the chaotic and quasi group's equations. The chaotic transposition was the third phase. It's involved in the logistic map and the plaintext in one equation to eliminate the plaintext characteristics. The Quasi transposition phase which is the fourth phase was used to shuffle the plaintext via quasi groups. The last phase was the substitution where the chaotic maps were used to transform the cipher values to be unreadable date.

The outcome results show that the proposed cryptosystem acquired outstanding performance figures, the proposed cryptosystem destroys any existing patterns in the input, and maximizes entropy. Moreover, the n-grams shows that the proposed cryptosystem was secure against the statistics analysis.

The results show that the block cipher mode gives higher entropy than the steam cipher mode, the entropy of the constant plaintext = 0, while the entropy of the ciphertext using stream cipher mode = 4.8580. In the Other hand the entropy of the ciphertext using block cipher mode was = 6.6736.

For ordinary plaintext, the entropy of the plaintext = 3.7600, even though the entropy of the ciphertext using stream cipher mode = 6.5466, but the entropy of the ciphertext using block cipher mode = 6.6506.

The result refers to encryption process use different keys, i.e. different quasi groups and different chaotic maps parameters, which means more and more randomness in the resulted ciphertexts.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this article.

Acknowledgments

We thank anonymous referees for their constructive comments.

References

[1] JiWon Yoon; Hyoungshick Kim. "An image encryption scheme with a pseudorandom permutation based on chaotic maps", Communications in Nonlinear Science and Numerical Simulation, 2010.

[2] Zbigniew Kotulski; Janusz Szczepański. Discrete chaotic cryptography. *Ann. Physik.* 1997, 6, 381-394.

[3] R. Schmitz; J. Franklin. Use of Chaotic Dynamical Systems in Cryptography. 2001, 338, 429-441.

[4] F. Anstett; G. Millerioux; G. Bloch. Chaotic cryptosystems: Cryptanalysis and identifiability. *IEEE Tran. Circuits and Systems I*, 2006, 53(12), 2673-2680.

[5] Fangjun Huang; Zhi-Hong Guan. Cryptosystem using Chaotic Keys. *Chaos Soliton Fractals*, 2005, 23(3), 851-855.

[6] Jean-Charles Faugère; Rune Steinsmo Ødegård; Ludovic Perret; Danilo Gligoroski. Analysis of the MQQ Public Key Cryptosystem. CANS. 2010, 169-183.

[7] E. Ochodková ans V. Snášel. "Using Quasigroups for Secure Encoding of File System", Proceedings of the International Scientific NATO PfP/PWP Conference ”Security and Information Protection 2001”, Brno, Czech Republic, 175-181,May, 2001.

[8] T. Ritter. “Orthogonal latin squares, nonlinear balanced block mixers,” Ritter Software Engineering Report, 1998.

[9] N.K. Pareek; Vinod Patidar; K.K. Sud. Discrete chaotic cryptography using external key. *Phys Lett A.* 2003, 309, 75-82.

[10] G. Álvarez; F. Montoya; M. Romera; G. Pastor. Cryptanalysis of a discrete chaotic cryptosystem using external key. *Phys Lett A*, 2003, 319, 334-339.

[11] N.K. Pareek a,b, Vinod Patidar A.; K.K. Sud. Cryptography using multiple one-dimensional chaotic maps. *Communications in Nonlinear Science and Numerical Simulation*, 2005, 10, 715-723.

[12] Jun Wei; Xiaofeng Liao; Kwok-wo Wong; Tsing Zhou. Cryptanalysis of a cryptosystem using multiple one-dimensional chaotic maps. *Communications in Nonlinear Science and Numerical Simulation*, 2007, 12, 814-22.

[13] Tao Xiang; Kwok-wo Wong; Xiaofeng Liao. An improved chaotic cryptosystem with external key. *Communications in Nonlinear Science and Numerical Simulation*, 2008, 14, 574-581.

[14] C. Kościenly. Generating quasi groups for cryptographic applications. *Int. J. Appl. Math. Computer. Sci.* 2002, 12(4), 559-569.

[15] D. Gligoroski; S. Markovski; S.J. Knapskog. Multivariate Quadratic Trap-door Functions Based on Multivariate Quadratic Quasi groups. In Proceedings of the American Conference on Applied Mathematics, (MATH08), Cambridge, Massachusetts, USA, 2008.

[16] M. Satti; S. Kak. Multilevel indexed quasigroup encryption for data and speech. IEEE Trans. on Broadcasting, 2009, 55, 270-281.

[17] A. Mileva; New developments in quasigroup-based cryptography. Multidisciplinary Perspectives in Cryptology and Information Security. IGI-Global, 2014, 286-317.

[18] S. Markovski. Design of crypto primitives based on quasigroups. *Quasigroups and Related Systems*, 2015, 23, 41-90.

[19] Y. Xu; H. Wang; Y. Li; B. Pei. Image encryption based on synchronization of fractional chaotic systems. *Communications in Nonlinear Science and Numerical Simulation*, 2014, 19(10), 3735-3744.

[20] M. Ahmad; I.R. Khan; S. Alam. Cryptanalysis of Image Encryption Algorithm Based on Fractional-Order Lorenz-Like Chaotic System. Emerging ICT for Bridging the Future. *Springer International Publishing Switzerland, AISC*. 2015, 338, 381-388.

[21] Entropy (information theory). Wikipedia. Wikimedia Foundation Inc. Available online: http://en.wikipedia.org/wiki/Entropy (accessed on 20 December 2017).

[22] Analysis, N-Gram. Available online: https://www.cryptool.org/en/cto-cryptanalysis/n-gram-analysis (accessed on 21 December 2017).

[23] N-gram. Wikipedia. Wikimedia Foundation Inc. Available online: http://en.wikipedia.org/wiki/N-gram (accessed on 21 December 2017).

[24] Autocorrelation. Wikipedia. Wikimedia Foundation Inc. Available online: http://en.wikipedia.org/wiki/Autocorrelation (accessed on 21 December 2017).