Text Encryption Algorithm Based on Chaotic Neural Network and Random Key Generator

Ghada Salim Mohammed
Dept. of Computer Engineering/ Madenat Al-Elem University/ College

Received in: 17/March/2016,Accepted in :18/October/2016

Abstract

This work presents a symmetric cryptography coupled with Chaotic NN , the encryption algorithm process the data as a blocks and it consists of multilevel(coding of character, generates array of keys (weights),coding of text and chaotic NN) , also the decryption process consists of multilevel (generates array of keys (weights),chaotic NN, decoding of text and decoding of character).Chaotic neural network is used as a part of the proposed system with modifying on it ,the keys that are used in chaotic sequence are formed by proposed key generation algorithm .The proposed algorithm appears efficiency during the execution time where it can encryption and decryption long messages by short time and small memory (chaotic NN offer capacity of memory), also the system uses secret keys with array of keys (weights of NN), that change at each iteration.

Keywords: Cryptography, Artificial Neural Network, Chaotic Neural Network.

Introductions

Cryptography and information security are considered as important sciences in the world, especially after using the computer in this science[1].The aim of cryptography is to make it impossible to take a cipher and reproduces the plain text without the corresponding key[2].Within the context of any application-to-application communication, there are some specific security requirements, which include **(Integrity, on-repudiation, Authentication and Privacy/confidentiality).**There are three types of cryptography (Secret key ,Public key and Hash functions) .[3]A neural network(NN) is a machine that is designed to model the way in which the brain performs a particular task. The network is implemented by simulated in software on a digital computer or using electronic components. Various ANN topologies were found. The most frequent ones are (Recurrent Neural Networks (RNN), General regression neural networks, Neural cryptography, Multilayer neural networks (MLP), and Chaotic Neural Network) [5].

Chaotic neural networks (CNN)

Chaotic system(CS) is statistically indistinguishable from randomness, and yet it is deterministic and not random at all, CS will produce the same outputs if given the same inputs. A random system will produce different outputs when it given the same inputs [6]. **CS** are sensitive to initial conditions, system parameters and topological transitivity . **CNN** offer greatly increase memory capacity, The network's features are high security and no distortion. **CNN** can be divided into three types:

- Location permutation algorithms scramble the locations of original data.
- Data value transformation algorithms transform the data value of the original data.
- The combining form performs Location permutation and Data value transformation operations **[7].**

Chaotic Iterations(CI)

Let us consider a system with a finite number $N \in N^*$of elements, so that each element has a boolean state. A sequence of length N of boolean states of the element corresponds to a particular state of the system. A sequence which elements belong to [1;N] is called a strategy. The set of all strategies in the system is denoted by S .The set B denoting {0,1} , $f : B^N \rightarrow B^N$ be a function .

To build a **CNN** using **chaotic iterations**, Let us reconsider the vectorial negation function denoted by($\boldsymbol{f_0 : B^N \rightarrow B^N}$)and its associated map F_{f0}: $[1;N] \times B^N \rightarrow B^N$. Where, for all (k, xx) $\in [1;N] \rightarrow B^N$, the response of the output layer to the input (k, xx) is F_{f0}(k, xx) [8]. The output layer can be connected to the input layer as it is illustrated in Figure (1)

The network is called **CNN** if its biases and weights of neurons are determined by CS. The encryption scheme belongs to the category of value transformation. Based on a binary sequence generated from the 1-D logistic map, the biases and weights of neurons are set in each iteration[9]**.**

The 1-D logistic map

$F_{f0}(S^0; xx^0)$**:** The proposed algorithm used the 1-D logistic chaotic map with neural network to produce a combination of CNN, based on a binary sequence generated from the logistic chaotic map. logistic chaotic map is**:**

xx(m+l) = U * xx(m)*(l-xx(m))   m=1 ...... N

where this equation evolve the CS xx(l), xx(2), ... , xx(N),that will be used to create b(0), b(l), ..., b(8N-1) . by the generating schema that 0.b(8m-8)b(8m-7) …. b(8m-2)b(8m-l) …is the binary form of xx(m) for m = 1, 2, ..., N.

The Proposed encryption algorithm

The proposed encryption algorithm is consists of the following steps, which are (1-coding process ,2- key generation and 3- ciphering using chaotic neural network) , Action steps of the proposed algorithm could be clarified as follows: (Figure (2): Illustrates the steps of proposed system).

The proposed encryption algorithm

Input: Plain text, keys

Output: The cipher text.

1. Begin.
2. Convert each character of plain text to special code by using coding algorithm .
3. Convert each value of character after coding to binary form (8-bits) .
4. Prepare the keys to generate the array of weights using generate key algorithm .
5. Divide the binary plain text to blocks (block with length 16-bits)and perform the following steps **(length of plain text/16)** times
   - Put the bits of block in 2-d array [4*4], create new 2-d array [4*4] by making the last column in this array to be the first row in new array ,the third column will be the last row, the second column will be the third row and finally the first column will be the second row in new array.
   - Convert the new 2-d array [4*4] to 1-d array then reverse the location of array.
   - Apply chaotic neural network by using the modified method algorithm .
   - Using the array of weights to generate the new array of weights for next block process by using new key algorithm .
   - **Next block.**
6. Print the cipher text to file.
7. **End.**

The proposed coding algorithm

This proposed algorithm coding the character of plaintext and transform the ASCII code value for each character to another value. the steps of this algorithm are:

Input: The character, key.

Output: the code of character(decimal value).

1. **Begin.**
2. Input key of coding
3. Input the character and convert it to ASCII code
4. Find **new code= ASC+KC+ASM**

Where (**ASC** is ASCII code of input character, **KC** is key of coding, **ASM** the value of character location **mod 8**).

5. **End .**

The proposed algorithm to generate array of key.

This proposed algorithm generate 2D- array of keys (the weights that are used in CNN) based on random integer numbers saved as a secret keys and the length of binary plain text ,all these are used with some logic gates to generate .(Figure (3): Illustrates the steps of proposed algorithm).

Input: BL: length of binary plain text, keys.

Output: the 2-D array of keys.

1. **Begin .**
2. Input the integer numbers **R1**, **R2** and use them to initial the generate key process, the prepare process of keys it is performed by :
   - Find the results of (K1=R1**mod**16, K2=R2 **mod**16).
   - Convert (K1,K2)decimal values to binary blocks (block 4-bits length).
   - Convert each block from 4-bit to 16 - bit length by:

∗ **(RK1=K1+(K1 XOR K2)),(RK2=K2+(K1 XOR K2)).**

∗ **(RK11=RK1+NOT(RK2)),(RK22=RK2+NOT(RK1))** (+ :is append between two blocks)

3. Find **(RES1= RK11+RK22)**, the resulted binary block with length 32-bits**.**
4. Divide **RES1** to 8 block **(s1,s2,s3,s4, s5,s6,s7,s8)**
5. Convert each binary block to decimal number

$$(\mathbf{S_1 \rightarrow DS_1}\ , \mathbf{S_2 \rightarrow DS_2}, \mathbf{S_3 \rightarrow DS_3}\ \ , \mathbf{S_4 \rightarrow DS_4},$$
$$\mathbf{S_5 \rightarrow DS_5}\ , \mathbf{S_6 \rightarrow DS_6}, \mathbf{S_7 \rightarrow DS_7}\ \ , \mathbf{S_8 \rightarrow DS_8}\ )$$

6. Find the results of

∗ Binary of (DS_1 mod 4),store the result 2-bits in locations ((0,0),(0,1 of key($w_{i,j}$).

∗ Binary of (DS_2 mod 4),store the result 2-bits in locations ((0,2),(0,3)) of key($w_{i,j}$).

∗ Binary of (DS_3 mod 4) ,store the result 2-bits in locations ((0,4),(0,5)) of key($w_{i,j}$).

∗ Binary of (DS_4 mod 4) ,store the result 2-bits in locations ((0,6),(0,7)) of key($w_{i,j}$).

∗ Binary of (DS_5 mod 4),store the result 2-bits in locations ((1,0),(1,1)) of key($w_{i,j}$).

∗ Binary of (DS_6 mod 4),store the result 2-bits in locations ((1,2),(1,3)) of key($w_{i,j}$).

∗ Binary of (DS_7 mod 4),store the result 2-bits in locations ((1,4),(1,5)) of key($w_{i,j}$).

∗ Binary of (DS_8 mod 4),store the result 2-bits in locations ((1,6),(1,7)) of key($w_{i,j}$).

∗ Convert the result 2-D array of key ($w_{i,j}$) to 1-D array then

- Replace between locations ((1,0),(3,2),(5,4),(7,6)).
- Take the locations (8 to 15) reverse it (15,14,13,12,11,10,9,8)

∗ Convert the 1-D back to 2-D array($w_{i,j}$) and use the last array as key for the encryption process of the first block of the plain text.

∗ Use this array as input to **New Key Algorithm(3.1.2.A)**to generate new array of key for the encryption process of the next block.

7. **End**

The proposed algorithm to new key algorithm

This algorithm aims to generate new array of keys (weights)by using the weights of previous iteration of CNN in some equations .

Input: previous array of key(weights).

Output: New array of key(weights.

1. **Begin .**
2. **apply the following equations:**
   a. **new($w_{i,j}$)= old ($w_{i,j}$)+XOR old ($w_{j,i}$)**
      and when i=j the following equation will apply:
   b. **new($w_{i,j}$)= old ($w_{i,j}$)+XOR old ($w_{i,j+1}$)**
      where i=0,1. j=0,1,2,3,4,5,6,7
3. **End.**

The proposed algorithm to encryption using chaotic neural network.

The result of this algorithm is a cipher code , the input text consists of finite number of state(character) each state converts to binary (boolean state) this first steps to chaotic iteration . by using a special function and its associated map, the results of the output layer will represent. the input binary blocks which represent the input to first layer in NN, then start the evaluation of chaotic sequence ($xx^{n+1}= F_{f0}\, b^n;\ xx^n) \in B^N$) is computed 1-D logistic map in the NN, This value of xx^1 is published as an output, xx^1 is sent back to the input layer, to act as boolean state in the next iteration ,also it is used to compute nn(k) the new inputs to first layer in NN the result from first layer which will be input to the second layer in with new weights and basis the output from this layer will be input to final layers to give the cipher code then begin new iteration .

Input: Block (block of 16-bits(**BLO**)), Specified key(**U** ,X(0),$w_{i,j}$).

Output: Cipher code

1. **Begin.**
2.

- Take the (0-7)bits from(**BLO**)and save a copy of it in **B1** (1-Darray with size 8) , Take the (8-15) bits from (**BLO**) and save a copy of it in **B2** (1-Darray with size 8)
  then calculate the layer one of NN :
  BL1(i)=B1(i) XOR W(0,j). where (i=0-7, j=0-7).
  BL2(i)=B2(i) XOR W(1,j). where (i=0-7, j=0 -7).

3. Convert the (0-7)bits of (**BLO**) to decimal value **DE1**and the (8-15) bits to decimal value **DE2.**(**BL1 :** the binary representation of **DE1**) ,(**BL2 :** the binary representation of **DE2**)
4. Using the specified key(U,XX(0)) to start the chaotic sequences:
   XX(l), X(2), ... , XX(LT/8) by XX(n+1)= U *XX(n)*(1-XX(n))
   (LT : is the length of plain text after coding) ,where for each iteration generate two elements of XX.
5. Create a chaotic bit sequences b(0), b(l), ..., b(7) … is the binary representation of XX(n), and b(8), b(9), ..., b(15) … is the binary representation of XX(n+1),
   For each iteration generate 16 location of(b()) from the binary representation of two generated elements of XX.
6. calculate the layer two of NN:
   - Find the value of g(n) by $\sum_{i=0}^{7} \mathrm{BL1(i)} * 2^{i}$
   - Find the value of gg(n+1) by $\sum_{i=0}^{7} \mathrm{BL2(i)} * 2^{i}$
   - Find the value of nn(k)=b(k) XOR W(i,j)
7.
   Where k= 0 to 15, i =0 to 1 , j=0 to 7.
   Find the value of ww(i,j)
   z=0 to 1, i= 0 to**7** , j=0 to 7

$$ww_{(i,j)}\begin{cases} 1 \ \ \mathrm{if\ i = j\ and\ nn(8z + \mathbf{i}) = 0} \\ -1 \ \ \mathrm{if\ i = j\ \ and\ \ nn(8z + \mathbf{i}) = 1} \\ \quad 0 \qquad\qquad \mathrm{if\ i \neq j} \end{cases}$$

- compute the biases :

$$\theta i\left\{\begin{array}{l} \frac{1}{2}\ \ \mathrm{if\ i = j\ and\ nn(8z + i) = 0} \\ -\frac{1}{2}\ \mathrm{if\ i = j\ \ and\ \ nn(8z + i) = 1} \end{array}\right\}$$

- Find the value of $\grave{\mathrm{d}}$(n) byj=0 to 7, ii=8 to 15

1) $\grave{\mathrm{d}}\mathrm{(n)}$ =f($\sum_{j=0}^{7} \mathrm{ww(i,j)} * \mathrm{BL1(i)} * 2^{i} + \theta i$).
2) $\grave{\mathrm{d}}\grave{\mathrm{d}}\mathrm{(n+1)}$ =f($\sum_{j=0}^{7} \mathrm{ww(i,j)} * \mathrm{BL2(ii)} * 2^{i} + \theta i$)

where f(x)=1 if x>0 otherwise f(x)=0
f(x) is a comparison function if the value(x=$\sum_{j=0}^{7} \mathrm{ww(i,j)} * \mathrm{BL1(i)} * 2^{i} + \theta i > 0$) so the result will be **1** otherwise the result will be **0**.

8. calculate the final layer of NN:
   - Find the value of ǵ(n) by $\sum_{i=0}^{7} \grave{\mathrm{d}}_{(n)} * 2^{i}$
   - Find the value of ǵǵ (n+1) by $\sum_{i=0}^{7} \grave{\mathrm{d}}\grave{\mathrm{d}}_{(n+1)} * 2^{i}$
   - Save ǵ (n),ǵǵ (n+1) as ciphered code.
9. **End.**

In deciphering by using chaotic neural network is the same steps of ciphering process (step 4 to step7) except of the input to algorithm of deciphering will be ǵ (n),ǵǵ (n+1) instead of g(n),gg(n).each iteration will cipher two characters from the original plain text(n refers to the number of character will process).the next section the deciphering process will explain in details.

The Proposed deciphering method

Action steps of the proposed system could be clarified as follows:

The proposed algorithm of deciphering process.

Input: Cipher text, keys.

Output: The plain text.

```
1. Begin.
2. Input the cipher text
3. Input the required keys and prepare the specific keys to generate the array of weights.
4. Generate the array of weights by using generate key algorithm .
5. Divide the binary cipher text to blocks each one with length 16 bits and perform the
   following steps (length of cipher text/16)times:
   • Begin
   • Apply chaotic neural network by using the modified method algorithm (the steps
     from 4 to step 7).
   • The two decimal values that result from the previous step will convert to binary
     (BL1,BL2) then apply :
       1) B1(i) = BL1(i)XOR W(0,j).   where (i=0-7, j=0-7).
       2) B2(i) = BL2(i)XOR W(1,j).   where (i=0-7, j=0 -7).
     then BLO =(B1 +B2). (BLO: is a block of 16-bits that stored in 1-D array)
   • Reverse the elements of array and convert the 1-D array to 2-d array [4*4] .
   • Convert the 2-d array [4*4] to new 2-d array [4*4] by making the first row in 2-d
     array be the last column in new 2-d array, the last row in 2-d array will be the third
     column in new 2-d array, the third row will be the second row and finally the first
     column will be the second column.
   • Using array of weights and generate the new array of key for next block process by
     using New Key Algorithm .
   • Next block
6. Divide the cipher text to blocks each one 8-bits length and convert it to decimal value.
7. Decoding each decimal value by using the decoding algorithm .
8. Print the plain text to file.
9. End.
```

The generating key algorithm and chaotic neural network are by using the modified method algorithm which are the same algorithms that applied in the ciphering process.

The proposed algorithm to decode the character of cipher text

Input: The decimal value

Output: The ASCII code of character (decimal value).

```
1. Begin .
2. Input the key of coding KC.
3. Input the decimal value(new code).
4. Find ASC = new code -KC-ASM.
   Where(ASC is ASCII code of input character, ASM is the value of character location
   mod 8).
5. Convert the ASCII code to the corresponding character.
6. End.
```

Implementation of the proposed system

Computer simulation and evaluation of the proposed random key generator

The statistical randomness tests that applied to binary sequences to test these sequences are:

1. **The Frequency Test**[2]

$$\chi^2 \frac{(\mathrm{N}_0-N_1)^2}{N} =$$

Where (N_0 : the number of occurrences of 0's in the N-bit sequence, N_1 : the number of occurrences of 1's in the N-bit sequence).

2. **The Serial Test**

$$\chi^2=[4/(N-1)\sum_{i=0}^{1}\sum_{j=0}^{1}N_{ij}{}^2-(2/N)(\sum_{i=0}^{1}N_i{}^2)]+1$$

where N_{00} , N_{01} , N_{10} and N_{11} be the number of occurrences of 00,01,10 and 11 , respectively in the N-bit sequence.

3. **The Poker Test**

$$\chi 2=(2^m/K)\sum_{i=0}^{2^m-1}(Y_i^2/C^m{}_i)-K.$$

where **K** : the number of blocks., **M** : the length of block ,$\boldsymbol{Y_i}$: the number of m-bit subsequences having i 1's and (m-i) 0's, and so on.

4. **The Autocorrelation Test**

The goal of this test is to check for correlations between the sequence **s** and shifted versions of it. Let **d** be a fixed integer, 1 <= d < =.$\lfloor {}^{n}/_{2} \rfloor$ The number of bits **is** not equal to their d-shifts is $\mathrm{A(d)} = \sum_{i=0}^{n-d-1} s_i \oplus \mathrm{s}_{\mathrm{i+d}}$.

The autocorrelation test equation involves the calculation of χ^2 using :

$$\chi^2 = \frac{\left(A(d)-\frac{n-d}{2}\right)}{\sqrt{n-d}}$$

- The value χ^2 for the frequency test must be less than or equal to 3.8415 to pass this test. The value $\chi^2$for the serial test must be less than or equal 5.9915 to pass this test. The value χ^2 for the poker test must be less than or equal 14.0671 to pass poker test. The valueχ^2 for the autocorrelation test since the value of A(d) for all are less than 1.96 .**[4]**

It is clear from Table (1) that all sequence of bits that tested pass the thresholds of statistical tests.

The execution time for the proposed system:

We tested the encryption and decryption algorithms on a system running on (CORE i7-4500 with 6.00 GBRAM, CPU 2.40 GHz,64-bit operating system), the algorithms of encryption and decryption applied for different messages with different sizes, below will illustrate the execution time for encryption and decryption process with example for massage before and after encryption process with keys value **(KC=9, R1=24, R2=38, U=2, X(0)=21).** The cipher message can be transmitted as blocks of binary or can be converted each 8-binary digits to decimal value then transmits it.

Conclusion

In this work , we used the chaotic neural network because chaotic system will not produce the same results if the inputs keys are not the specified keys so it is difficult to attack to have the plain text without knowing the secret keys, in this paper to increase the security we use many algorithms to cipher the data as a secret key cryptography, these algorithms are based on(position permutation and value transformation) where the coding and decoding of character algorithms will transform the value to another value .the proposed generating key(weights) algorithm will be based on scramble the positions of data and transformed it, the resulted data from this algorithm will be used as weights in encryption and decryption process in addition to the weights that are generated by chaotic neural network . finally all these algorithms and modifications will give the proposed system the efficiency and the security ,this appears from applying randomness test on the ciphering system.

References

1. Rolf ,O. (2005)"Contemporary Cryptography", Artech House , Boston ,London.

2. Michal Janosek; Eva Volna; Martin Kotyrba and Vaclav Kocian.(2012).“Cryptography based on Neural Network". Proceedings 26th European Conference on Modeling and Simulation, www.scs-europe.net/conf/ecms.

3. Behrouz, A. Forouzan.(2007).“Cryptography and Network Security”, Tata McGraw-Hill.

4. Smid ,M. E. and Branstad, D. K.(1988).“The Data Encryption Standard:Past and Future”. Proceedings of the IEEE, 76, 5, 550-559.

5. Amr, H. Yassin ;Adel. El-Zoghabil and Hany, H. Hussien.(2013)." Survey Report on Cryptography Based on Neural Network", An International Journal of Emerging Technology and Advanced Engineering Website: www.ijetae.com .

6.Tope Komal; Rane Ashutosh; Rahate Roshan and Nalawade, S.M..(2015)." Encryption and Decryption using Artificial Neural Network", An International Advanced Research Journal in Science, Engineering and Technology.

7. Kamila ,N. K.; Haripriya Rout and Nilamadhab Dash.(2014)." Stego- Cryptography Using Chaotic Neural Network ",American Journal of Signal Processing..

8.Jacques ,M. Bahi; Christophe Guyeux, and Michel Salomon.(2011)."Building a Chaotic Proved Neural Network".

9. Adel, A. El-Zoghabi; Amr,H. and Yassin, Hany H. Hamdy.(2014)."Public key Cryptography Based on Chaotic Neural Network",International Journal of Computer Application.

Table (1) Statistical tests results for the different length sequences .

The file size	Frequency Test	The serial test	The poker test M=6	The Autocorrelation Test
320-bits	0.2	0.310	1.516	0.5434
640-bits	0.4	0.506	3.657	0.7637
960-bits	0.6	0.704	4.972	0.9334
1280-bits	0.8	0.903	6.735	1.0766
1500	1.176	1.179	10.613	1.3443

Table (2) The execution time for different messages

The number of characters of message	The time of execution of encryption process	The time of execution of decryption process
137	1msec.	1msec.
500	1msec.	1msec.
1000	3msec	3msec
1500	4msec	4msec

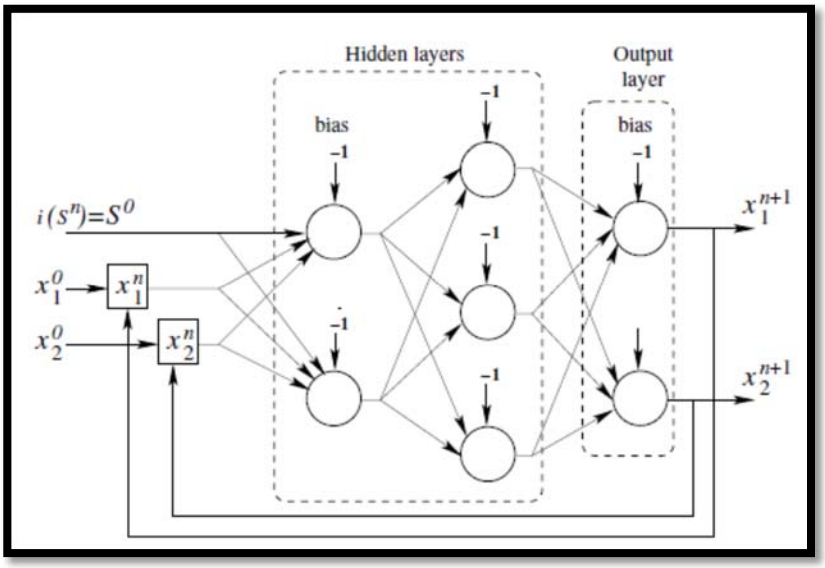
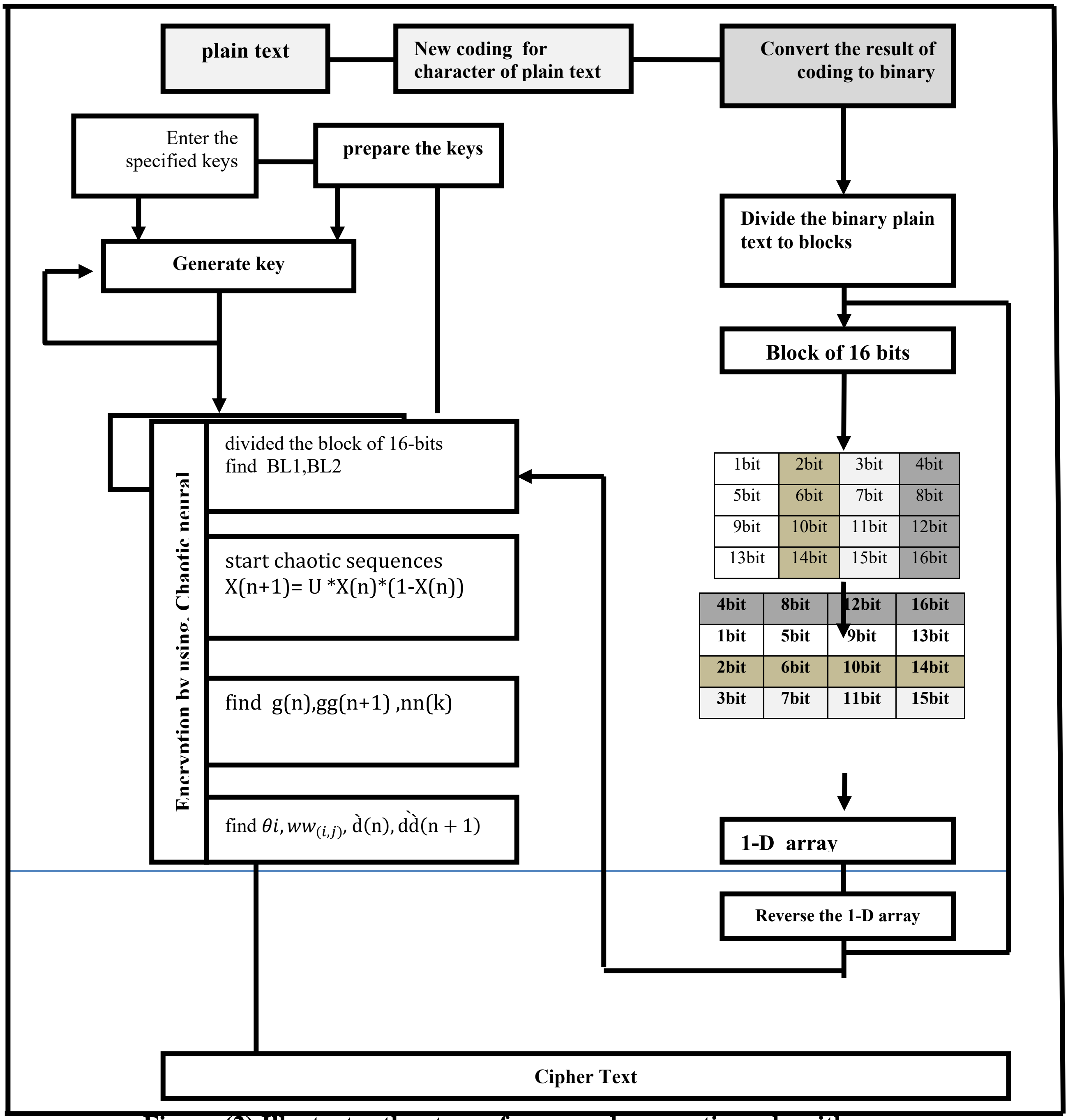


Figure (1) Illustrate the output layer connected to the input layer in CNN


plain text
New coding for character of plain text
Convert the result of coding to binary
Enter the specified keys
prepare the keys
Generate key
Divide the binary plain text to blocks
Block of 16 bits
Encryption by using Chaotic neural
divided the block of 16-bits find BL1,BL2
start chaotic sequences X(n+1)= U *X(n)*(1-X(n))
find g(n),gg(n+1) ,nn(k)
find θi, ww(i,j), d̀(n), dd̀(n + 1)
1-D array
Reverse the 1-D array
Cipher Text

1bit	2bit	3bit	4bit
5bit	6bit	7bit	8bit
9bit	10bit	11bit	12bit
13bit	14bit	15bit	16bit

4bit	8bit	12bit	16bit
1bit	5bit	9bit	13bit
2bit	6bit	10bit	14bit
3bit	7bit	11bit	15bit

Figure (2) Illustrates the steps of proposed encryption algorithm

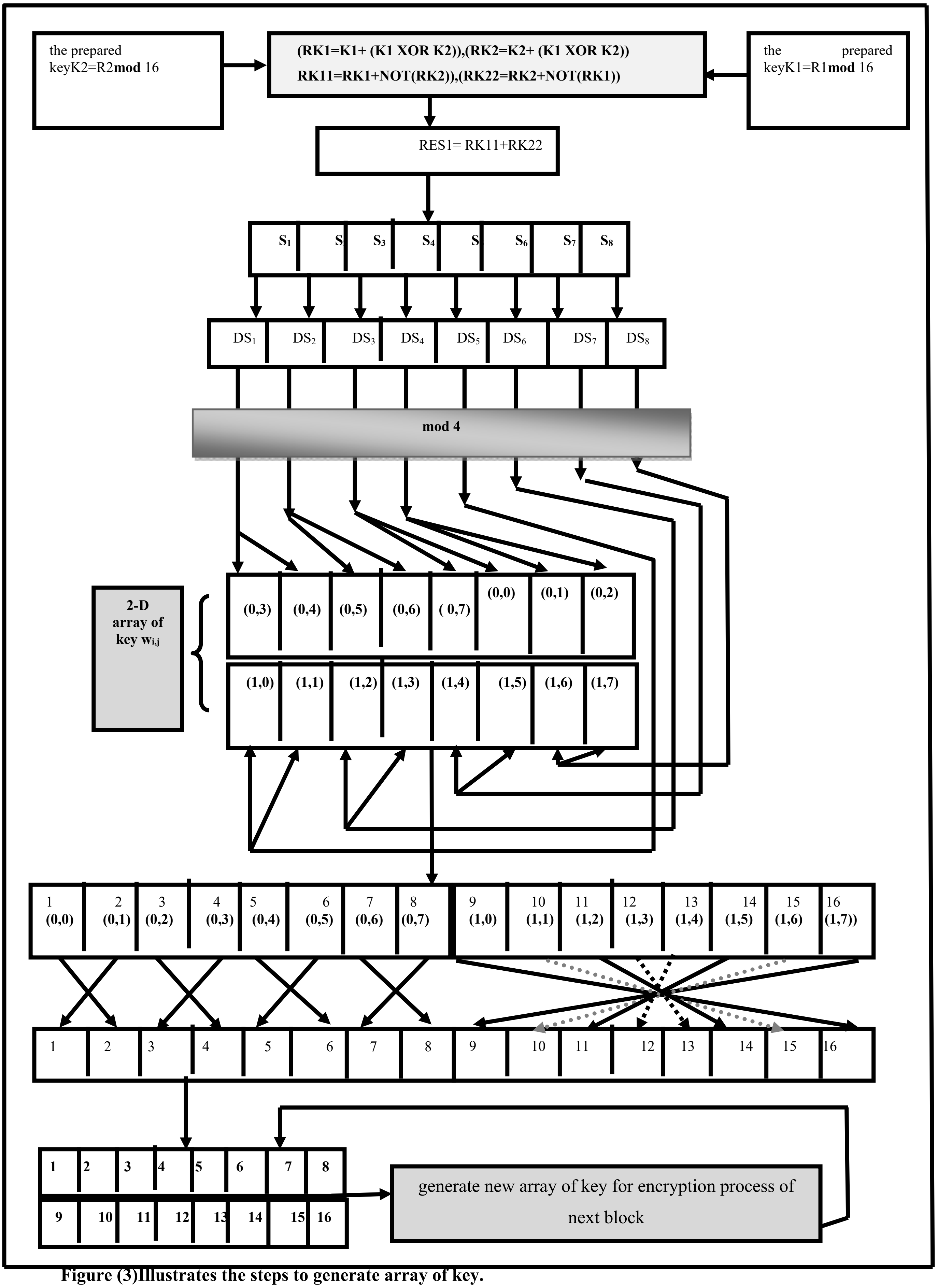


Figure (3)Illustrates the steps to generate array of key.

خوارزمية تشفير نص تعتمد على الشبكة (Chaotic)ومولد مفاتيح عشوائي العصبية

غادة سالم محمد
قسم هندسة الحاسوب/ كلية مدينة العلم الجامعة

استلم في: 17/اذار/2016,قبل في :18/تشرين الاول/2016

الخلاصة

في هذا العمل تم تمثيل استعمال نظام التشفير باستعمال المفتاح السري مع الشبكة العصبية Chaotic ,خوارزمية التشفير تعالج البيانات بشكل مقاطع وتشمل عدة مستويات (ترميز حرف، وتوليد مصفوفة المفاتيح (الأوزان)، ترميز النص، Chaotic NN)، أيضا عملية فك التشفير تشمل عدة عمليات تتضمن (توليد مصفوفة المفاتيح (الأوزان) Chaotic NN، فك النص، فك الحرف) تستخدم الشبكة العصبية Chaotic كجزء من النظام المقترح مع بعض من التعديل عليها وتتولد المفاتيح التي تستخدم في Chaotic NNمن خوارزمية توليد المفاتيح المقترحة. النظام المقترح اظهر الكفاءة من خلال وقت التنفيذ حيث يمكن تشفير وفك تشفير رسائل طويلة خلال وقت قصير وذاكرة صغيرة (chaotic NN يوفر من سعة الذاكرة). كذلك فأن النظام يستخدم مفاتيح سريه ومصفوفة مفاتيح (أوزان للشبكة العصبية) التي تتغير في كل دوره مع كل مقطع.

الكلمات المفتاحية: التشفير ،الشبكات العصبية الذكية, فك التشفير.