Chaotic neural network based pseudo-random sequence generator for cryptographic applications

Conference Paper · October 2015

DOI: 10.1109/ICATCCT.2015.7456845

2 authors, including:

Manjunath R Kounte
Reva University
68 PUBLICATIONS **403** CITATIONS

SEE PROFILE

Chaotic Neural Network Based Pseudo-Random Sequence Generator for Cryptographic Applications

Lokesh S
PG Student, Dept. of Electronics & Communication
Reva Institute of Technology & Management, Bangalore
lokesh.s.knp@gmail.com

Manjunath R Kounte
Assistant Professor, Dept. of Electronics & Communication
Reva Institute of Technology & Management, Bangalore
manjunath.kounte@gmail.com

***Abstract*--the goal of any cryptographic system is the exchange of information among the intended users without any leakage of information to others who may have unauthorized access to it. A common secret key could be created over a public channel accessible to any opponent. As we reviewed in this paper, a cryptographically efficient pseudorandom sequence should have the characteristics of high randomness and encryption effect. Neural networks can be used to generate common secret key. In case of neural cryptography, both the communicating networks receive an identical input vector, generate an output bit and are trained based on the output bit. The two networks and their weight vectors exhibit a novel phenomenon, where the networks synchronize to a state with identical time-dependent weights. The statistical quality of pseudo-random sequences determines the strength of cryptographic system. The generation of pseudo-random sequences with high randomness and encryption effect is a key challenge. We also discuss simple examples of neural network.**

Index Terms—Neural cryptography, Chaotic, Pseudo-random sequence generator

I. INTRODUCTION

Nowadays information security has become an important aspect in every organization. In other words, the people have to be assured that the information to be read by only the sender and receiver. The basic need to provide security is using cryptography. The information is distributed, shared and exchanged at a very fast rate over the open wired/wireless networks. Therefore, it demands the need of methods to ensure confidentiality and privacy of sensitive information asset before transmitting it through communication channel.

Cryptography can be defined as the exchange of data into a mingle code that can be deciphered and sent across a public or private network. Cryptography is the practice and study of hiding information. It is a critical part of secure communication. Cryptograph not only protects data from robbery or alternation but can be used as well for user authentication. Cryptography has two core styles of encrypting data symmetrical and asymmetrical. Symmetric encryptions use the same key for encryption and decryption process, and also can be defined as a secret-key, shared-key, and private-key [1]. Asymmetric cryptography uses different encryption keys for encryption and decryption process. In this case an end user on a network, public or private, has a pair of keys; one for encryption and one for decryption. A system that provides encryption and decryption is referred to as a cryptosystem and can be created through hardware components or program code in an application.

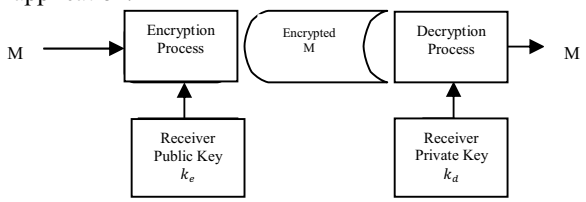

Figure 1: Cryptographic Components

Chaos theory studies the behavior of dynamical systems, whose state evolves with time. Small differences in initial conditions yield widely diverging outcomes for such dynamical systems, rendering long-term prediction impossible in general. This happens even though these systems are deterministic, meaning that their future behavior is fully determined by their initial conditions, with no random elements involved. In other words, the deterministic nature of these systems does not make them predictable.

Chaotic systems possess characteristics which are suitable for cryptographic use such as their sensitivity to initial conditions, system parameters, Ergodicity, mixing properties, unstable periodic orbits with long periods and random-like behavior [10]. Due to these features, chaotic systems are extensively incorporated into the encryption systems to ensure secure transmission of text, images, audio or video over insecure networks. Chaos based cryptosystems are known to have yielded better results and provide high levels of security than the conventional encryption techniques.

The structure of rest of this paper is as follows: Literature review of neural network based cryptographic pseudo random sequences is described in section II, while the proposed work is described in section III. Section IV we discuss result of proposed Generator. Open Research issues in V. While the conclusion of the work is drawn in section VI.

II. LITERATURE REVIEW

There has been an increasing interest in the application of different classes of neural networks to problems related to cryptography in the past few years. Much cryptography methods are available which are based on number theory but it has the disadvantage of requirement a large computational power, complexity and time consumption.

A. Neural Cryptography

Wolfgang Kinzel proposed a secret key over a public channel using artificial neural networks. The artificial neural

network contains of two multi-layer neural networks trained on their mutual output bits and able to synchronize. The two networks starting from random initial weights and learning from each other with two multilayer networks relax to the state with time dependent identical synaptic weights. The partners didn't exchange any information over a secret channel before their communication. Synchronization of neural networks can be considered as the key generation in cryptography. The common identical weights of the two partners can be used as a key for encryption. The neural cryptography is the first algorithm for key generation over public channels which are not based on number theory. Experimental result shows that the model is fast, simple, and secures [2].

B. New Security on Neural Cryptography with Queries

N. Prabakaran proposed a secret key using neural cryptography, based on synchronization of Tree Parity Machines (TPMs) by mutual learning. The system has two identical dynamical systems, which starting from different initial conditions and synchronized by a common input values which are coupled to the two systems. The networks received a common input vector after calculating their outputs and updated their weight vectors according to the match between their mutual outputs in every time step. The input or output relations are not exchanged through a public channel until their weight vectors are matching and can be used as a secret key for encryption and decryption of secret messages. The weight vectors of the two neural networks begin with random numbers, which are generated by Pseudo-Random Number Generators (PRNGs). The proposed model fixed the security against numerical attacks [3].

C. Design of an efficient neural key generation

R. M. Jogdand proposed a common secret key generated based on neural networks. The neural cryptography has two communication networks that received an identical input vector, generated an output bit and are trained based on the output bit. The network model initials the weight randomly and the input object is generated by another source and the outputs bit are generated finally and exchange between patterns. The weight may be modified if the outputs of both partners are matched. The modified weight after synchronize act as a secret key for the encryption and decryption process. Simulation results show that the cryptosystem based on ANNs is secure [4].

D. Cryptanalysis of a cryptographic scheme based on delayed Chaotic neural networks

Jiyun Yang analyzed the proposed model of Wenwu Yu [9]. It was difficult to obtain the key of Yu et al.'s cryptosystem through classical attacks because of large key space. However, as the same key stream is used in every encryption process, it can be easily obtained by the chosen plaintext attack using two pairs of plaintext and cipher texts only. Simulation results show that the proposed chaotic cryptography is insecure [5].

E. LFSR and PLA based Complex Code Generator for Stream Cipher

Programmable long PN sequence generator using Linear Feedback Shift Register is presented that uses different taps randomly or in a predefined manner. Various statistical tests were also performed which show random nature of the sequence generated. Further, the sequence generated was applied to encrypt an image. The encrypted image shows white noise characteristics [9].

F. An Empirical Investigation of Using ANN Based N-State Sequential Machine and Chaotic Neural Network in the Field of Cryptography

Nitin Shukla proposed two artificial neural networks for cryptography. The First network is neural network based n-state sequential machine and other one is chaotic neural network. The first network generated a finite state sequential machine using simple recurrent neural network based on back propagation training algorithm. The starting state of the n-state sequential machine can be used as a key for encryption and decryption process. The second network divided the message into blocks and identified the initial value, and the control parameter, then generated the chaotic sequence. The weights and biases of the neural network are determined based on the chaotic sequence, and act as a key for encryption and decryption process. Experimental results show that the two networks are secure, without any results about efficiency [6].

G. Artificial neural network based chaotic generator for cryptology

Ilker D. proposed a chaotic cryptosystems based on artificial neural network and chaotic generator synchronization. The ANN model generated the chaotic dynamics by the numerical solution of Chua's circuit. The proposed model have three initial conditions and time variable as input with two hidden layers and three chaotic dynamics as output. Many simulations are done on the number of neurons on hidden layer to find the best ANN structure obtained chaotic dynamics.

The chaotic dynamics act as the key for encryption and decryption process. The ANN model does not have any synchronization problem. The difference between the chaotic dynamics can be considered as an advantage of the ANN based chaotic generator. The major weaknesses of analogue circuit and the numerical solution of chaotic circuit are eliminated. Simulation results shown that the model are efficiency, secure, and can be applying on real time application [7].

III. FRAMEWORK BY PRATEEK SINGLA ET.AL

In the proposed generator, the neural network shown in Fig.2 is used, which is composed of four layers: the input layer, the first hidden layer, the second hidden layer and the output layer [8]. The input to the neural network is a 64-bit key M = [m1 m2….m64]. The output of the input layer is defined as eqn (1),

$$P = F^{n_0}(\sum W_0 M + B_0) \quad \ldots 1$$

Where,

$W_0 = [W_{0,0}, W_{0,1}, \ldots W_{0,7}, W_{1,0}, \ldots W_{7,7}]$ is of size 8×8,

Bias matrix of $B_0 = [b_0, b_1 \dots b_7]$, control parameter matrix

$G_0 = [g_0, g_1, \dots g_7]$ is of size 8×1,

To calculate the output of the input layer C, first the inputs $P_1, P_2 \dots P_{64}$ are multiplied by their respective weights $(W_{0,0}, W_{0,1}, \dots W_{0,7}, W_{1,0,} \dots W_{7,7})$ and added with the respective biases $(b_0, b_1 \dots b_7)$ of neurons. Then this value is used as current state x, and generated control parameter q to iterate the PWLCM map for n_0 times

Where n0 is a random number ($1 \leqslant n_0 \leqslant 10$) generated by the key generator and F (.) is the transfer function, which is the piecewise linear chaotic map (PWLCM). The PWLCM is among the most studied chaotic systems. Its system equation is described as follows:

$$s(t+1) = F(s(t), g) = \begin{cases} \frac{s(t)}{g}, & 0 < s \leq g \\ \frac{1-s(t)}{1-g}, & g < s(t) < 0 \end{cases} \quad \dots 2$$

Where x (k) is state of the map and q is the control parameter and satisfies 0 < x (k) < 1, 0 < q < 1.

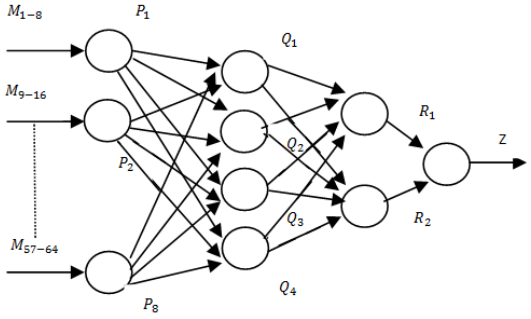

Fig2: Neural Network

After one complete operation of the chaotic neural network, the output of each neuron of the Input Layer directly becomes the input of that neuron for the next operation, apart from becoming the input to the first hidden layer for the same operation. The control parameter matrices G_i are transformed using the networks outputs in such a way that limit their range in [04, 0.6] as:

$$G_0 = (0.2 \times P) + 0.4 \quad \dots 3$$

This transformation is done to keep the value of the control parameters close to 0.5 in order to obtain the best chaotic behavior from the chaotic transfer function. Z (j) is the output of the neural network after jth operation and satisfies 0 < Z < 1. Z is normalized in the range 0-255 as in eqn (4):

$$W(j) = (Z(j) \times 10^{10}) \bmod (256) \quad \dots 4$$

If it crosses the threshold value of 127, then the next member of the pseudo-random binary sequence is taken as 1, else 0 given by eqn (5).

$$\psi(j) = \begin{cases} 0, & w(j) \leq 127 \\ 1, & w(j) \geq 128 \end{cases} \quad \dots 5$$

Key Generator,

The key generator uses the 1-D chaotic cubic map and Accepts a 64-bit key K $= K_1 K_2 K_3 K_4$, where K_i is a 16-bit Component of key K, to calculate the initial condition

$$v(0) = (\sum (K_i / 2^{16})) mod (1) \quad \dots 6$$

Of cubic map and returns the values y (n) on iterations. The state of the cubic map is governed as:

$$V(n+1) = \mu . v(n) . (1 - v(n) . v(n)) \quad \dots 7$$

Where μ= 2.59 is map's control parameter and the state Of the map y (n) satisfies 0 ≤v (n) ⩽ 1.

Proposed Algorithm

The steps of the proposed chaotic neural network based pseudo-random binary sequence generation are as follows:

1. Input key K to the key generator and iterate it 50 times and discard the values.
2. Again, iterate the key generator to initialize W0, W1, W2, W3, B0, B1, B2, B3, G0, G1, G2, G3, n0, n1, n2 & n3.
3. Provide the 64-bit seed P to the neural network.
4. Operate the neural network to obtain the output Z.
5. Normalize the output Z in the range 0-255.
6. Apply thresholding with 127 to extract next member (j) of the random binary sequence.
7. Assign G0 to P, G1 to Q, G2 to R, and G3 to Z after applying the transformation.
8. Repeat steps 3 to 7 to obtain the pseudo-random sequence of desired length.

To verify the randomness performance of proposed pseudo-random sequence generator, the randomness tests of Statistical Test Suite (STS) by NIST [11] are performed. The tests such as Frequency, Block frequency, tests are performed on the generated binary sequence.

Also statistical performance against the auto-correlation function, 0/1 balancedness, randomness and encryption efficiency of the proposed pseudo-random sequence generator is investigated.

The two keys of 64-bits used for simulation are K and P. The key space of proposed generator is

$$(2^{16}) \times (2^{16}) = 2^{128}$$

IV. Results & Discussion

In this section we discuss the results obtained in construction of the generator on Neural Network tool box in MATLAB.

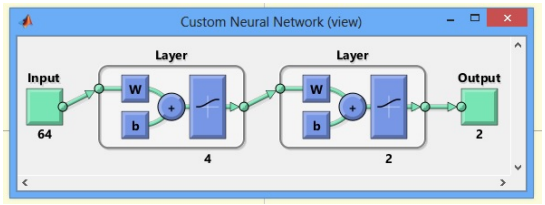

Fig 3: Neural Network

The above Neural Network accepts 64bit input generated by the key generator and produces a 64 bit pseudo random number. It also consists of 2 hidden layer neurons, 1st hidden layer with 4 neurons and 2nd hidden layer with 2 Neurons.

To test the encryption performance of the generated sequence, it is experimented with same image chosen in [9], i.e. standard 8-bits gray-scale Lena image of same size 256×256, shown in Fig. 4. The generated pseudo-random

sequence is xor'ed with the plain image bit stream to obtain encrypted image shown in Fig. 5. The histogram of the plain image is shown in Fig. 6, and that of the encrypted image is shown in Fig. 7.

Fig 4: Plain Image

The output of neural network 64bit is xored with the grey scale converted plain image to obtain an encrypted image has shown below. The average of the gray-level intensities of the encrypted image (Fig. 5) comes out as 180.5689, which is having an Information Entropy of 7.9970, which is nearly equal to ideal value 8.

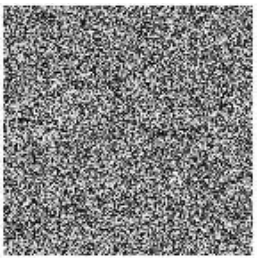

Fig 5: Encrypted plain image

This encrypted image has a random like behavior, which is difficult to decrypt by the opponent. Image histogram is a type of histogram that acts as a graphical representation of the tonal distribution in a digital image. This histogram is a graph showing the number of pixels in an image at each different intensity value found in that image. By looking at the histogram for a specific image a viewer will be able to judge the entire tonal distribution at a glance. Histogram of the plain image is shown in the fig below.

Grey scale digital image is an image in which the value of each pixel is a single sample, that is, it carries only intensity information. Images of this sort, also known as black-and-white, are composed exclusively of shades of gray, varying from black at the weakest intensity to white at the strongest.

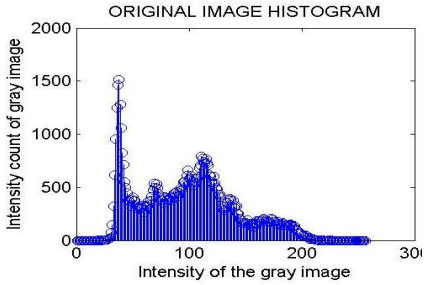

Fig 6: Pixels gray value distribution in plain-image

The histogram plots the number of pixels in the image (vertical axis) with a particular brightness value (horizontal axis). Also histogram of encrypted image is shown in the fig 7 below.

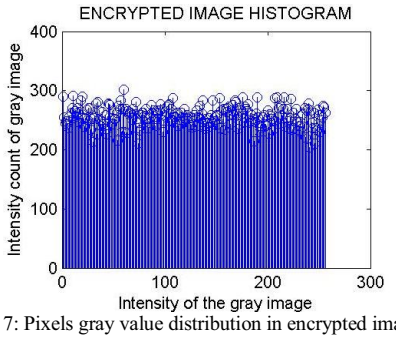

Fig 7: Pixels gray value distribution in encrypted image

This property shows that encrypted image pixel gray value is totally random having a value of 127.0653 which is lesser then the ideal value 127.5. Correlation co-efficient of the encrypted image as value 0.0034 which is near to zero.

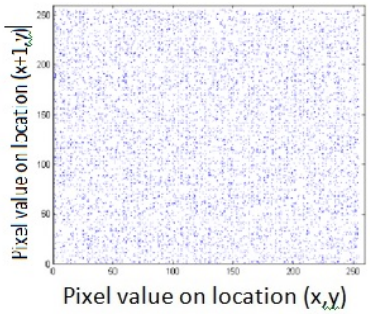

Fig 8: Correlation of adjacent pixels in encrypted image

Correlation of adjacent pixels shows that there is a complete randomness between the pixel values in the encrypted image fig 8. Frequency Deviation in Pixel Intensity 16.618 which is nearly equal to 15.7191, which is a LFSR based method [9]. Correlation co-efficient of the encrypted image calculated, which is found to be having the value has 0.0034, which is nearly equal to the 0.

Auto-Correlation Max Value of the adjacent pixels in encrypted image is found to be 0.00375 which is nearly equal to the idea value 0.

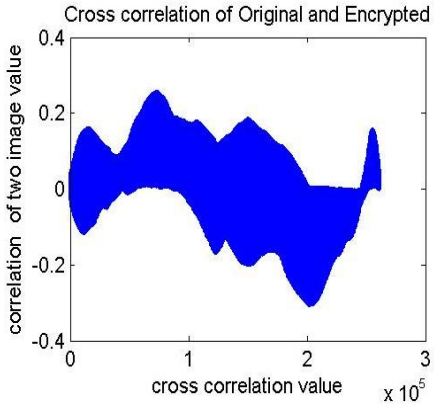

Fig 9: Cross Correlation of Original & Encrypted Image

Above figure 6.7 shows the cross correlation of original image with the encrypted image. This shows that the pixel values are random.

Any noise like sequences should have the property that sequence has to have an equality distribution property. Hence equal number of 0's and 1's has to be there in pseudo random sequence. Therefore to check whose property number of ones are counted and tabulated for particular set of values. Equality distribution graph shown in the fig10, which shows the measurement having nearly 50% of 1's. So the generator proposed satisfies the equality distribution.

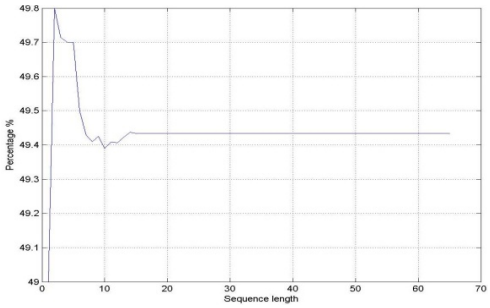

Fig 10: Equality distribution showing percentage occurrence of 1's

Maximum autocorrelation value of the generated sequence is found to be 0.00375, which is nearly equal to zero, for perfect noise.

Also some Random tests has been performed on the pseudo random sequence generated those are, Run Test, Approximate Entropy test [11]. Run Test, Approximate Entropy test has a value of 0.9242 & 0.5845.

V. Open Research Issues

A cryptographically efficient pseudo-random sequence should have the characteristics of high randomness and encryption effect. The statistical quality of pseudo-random sequences determines the strength of cryptographic system. The generation of pseudo-random sequences with high randomness and encryption effect is a key challenge.

VI. Conclusion

An efficient chaotic neural network based pseudo-random sequence generator is proposed. The generator utilizes the high sensitivity and randomness property of chaotic map such as the cubic map and the piece-wise linear chaotic map, coupled with the nonlinear complexity of a four-layer neural network. This also identifies new directions and open problems in pseudo-random sequence generator.

References

[1] William Stallings, "Cryptography and Network Security: Principles and Practice", (5th Edition), Prentice Hall, 2010.

[2] Wolfgang Kinzel, IdoKanter, "Neural Cryptography ▪ , Proceedings TH2002 Supplement", Vol. 4, 147 – 153, 2003.

[3] N. Prabakaran, P. Vivekanandan, "A New Security on Neural Cryptography with Queries", Int. J. of Advanced Networking and Applications, Vol. 2, Issue. 1, 437-444, 2010.

[4] R. M. Jogdand, Sahana S. Bisalapur, "Design of an efficient neural key generation", International Journal of Artificial Intelligence & Applications (IJAIA), Vol.2, No.1, 60- 69, 2011.

[5] Jiyun Yang, Xiaofeng Liao, Wenwu Yu, Kwok-wo Wong, Jun Wei, "Cryptanalysis of a cryptographic scheme based on delayed chaotic neural networks", Chaos, Solitons& Fractals, Vol. 40, Issue.2, 821-825, 2009.

[6] Nitin Shukla, Abhinav Tiwari, "An Empirical Investigation of Using ANN Based N-State Sequential Machine and Chaotic Neural Network in the Field of Cryptography", Global Journal of Computer Science and Technology Neural & Artificial Intelligence, Vol. 12, Issue.10,No. 1,17-26, 2012.

[7] Ilker Dalkiran, Kenan Danis, "Artificial neural network based chaotic generator for cryptology", Turk J Elec Eng& Comp Sci, Vol.18, No.2, 255- 240, 2010.

[8] Prateek Singla, Payal Sachdeva, Musheer Ahmad, A Chaotic Neural Network Based Cryptographic Pseudo-Random Sequence Design, 4th International Conference on Advanced Computing & Communication Technologies 8 Feb 2014.

[9] Farah Maqsood,Omar Farooq,Wasim Ahmad,"LFSR and PLA based Complex Code Generator for Stream Cipher" International Conference on Multimedia, Signal Processing and Communication Technologies 14 march 2009.

[10] G. Alvarez and S. Li,"Some basic cryptographic requirements for chaos based cryptosystems", International Journal of Bifurcation and Chaos, vol. 16, no. 8, pp. 2129-2151, 2006

[11] A. Rukhin, et al. "A Statistical Test Suite for Random and Pseudo-random Number Generators for Cryptographic Applications", NIST Special Publication 800-22, 2001.

[12] Manjunath R Kounte, Dr. B K Sujatha, "Top-Down Approach for Modelling Visual Attention using Scene Context Features in Machine Vision", International Journal of Applied Engineering Research, ISSN 0973-4562 Volume 10, Number 12 (2015) pp. 31585-31594.

[13] Manjunath R Kounte, Dr. B K Sujatha, "Identification of Visual Attention Regions in Machine Vision Using Saliency Map", 4th IEEE International Conference on Communication and Signal Processing-ICCSP-2015, Melmaruvathur, Tamilnadu, India, 2-4 April 2015.

[14] Divya S, Manjunath R Kounte "Implementation of Testing and simulation of PIM-SM Multicast for Ship Data Network" Proceedings of the Twelfth International Conference on Wireless and Optical Communications Networks (WOCN) ,Bangalore, Karnataka, India, 9-11 Sep. 2015.

[15] Manjunath R. Kounte, Dr. B. k. Sujatha, "Bottom up Approach for Modelling Visual Attention Using Saliency Map in Machine vision: A Computational Cognitive Neuroscience Approach", International Journal of Applied Engineering Research, volume 10, Number 11,2015, pp. 30153-30166.