A Chaotic Real-time Cryptosystem using a Switching Algorithmic-based Linear Congruential Generator (SLCG)

Raymond S. T. Lee and Henry W. S. Lam

Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Hong Kong

Summary

In order to increase the security level of the existing secure communication techniques used on top of a non-guarantee protocol like RTP, a chaotic real-time cryptosystem using Linear Congruential Generator with Switching Algorithm (SLCG) was introduced. This cryptosystem applied the technique of Chaos Theory for data synchronization during real-time data encryption and decryption. SLCG, as compared to traditional chaos generator such as Logistic Equation (LE), showed a promising result in terms of computation time, trajectory distribution and key length. In summary, the proposed cryptosystem saved about 16% in term of the total time for real-time data encryption and decryption. Besides, its trajectory distribution can be varied with respect to the changing key. This new cryptosystem increases the number of chaotic equations used and shows a significant improvement in terms of data encryption (and decryption) efficiency as well as security level.

Key words:

Chaotic Real-time Cryptosystem, Switching-based Linear Congruential Generator, Chaotic Data Synchronization

1. Introduction

1.1 Background

The word "Chaos" has appeared since 800 BC and was derived from the Greek $\chi\alpha o\varsigma$, which means a complete absence of order.

However, in Chaos Theory, "Determinism" is one of the basic principles. "Determinism" is the belief that every action happens is the result of preceding actions. Because of this deterministic nature, people can adopt Chaos Theory to describe the phenomenon of events happen in our world of experience. Although most of the events happened (and observed) in our world of living are highly complicated and almost seem to be totally unpredictable, it is believed that we can still model these phenomena using certain highly nonlinear (but deterministic) system – Chaos systems [1][8][12][14][18].

In the past decades, many researchers started to adopt Chaos Theory into cryptosystems [2][3][7][16][20][21]. The reason of applying chaotic algorithms on data encryption because of two intrinsic characteristics of chaotic algorithms: 1) Highly complex and nonlinear behaviors; 2) Sensitive dependence on initial conditions.

1.1.1 Highly complex and nonlinear behaviors

Owing to the highly complex and nonlinear signals generated by chaos generator, when it is imposed onto a data-stream, the output signal stream will become highly chaotic and seems to be totally unpredictable which increase the difficulty for hackers to analyze or even to comprehend these data-stream! So chaos signal generators are commonly adopted as so-called "signal masks" for data encryption, especially for real-time data encryption.

1.1.2 Sensitivity dependence on initial conditions

Another remarkable feature of chaos systems is their sensitivity to initial conditions. In other words, for a typical chaos system, even when there is a very minor variation in the initial conditions, the system dynamics will be varied tremendously. Based on this concept, when a chaotic data generator is used as the key(s) in a cryptosystem, this system sensitivity to initial conditions will substantially increased the complexity for a hacker to guess (and hack) the system.

1.2 Contemporary Chaotic Cryptosystems

In general, there are mainly two kinds of chaotic ciphers:

- Secure communications or cryptosystems based on chaos synchronization technique of analog circuits [6][9][13][17][19];
- Chaos-based ciphers realized on digital circuits or computers with finite precision effect [10][11].

In addition, the use of chaos in some other areas can enrich the knowledge about the design and the performance analysis of chaotic ciphers. They are: chaotic communications (especially the chaotic spread spectrum communications), chaotic pseudo-random number generations, chaotic signal estimation and detection, and chaotic digital watermarking [12][15].

However, it was an open question that building a chaotic cryptosystem on top of a non-guarantee protocol like RTP [4]. In fact, most of the real time multimedia applications transfer data using the non-guarantee protocols like UDP. These kinds of applications allow packet loss during data transmission. It becomes the main problem of synchronization of chaotic cryptosystem. The chaotic sequence will not be synchronized due to packet loss during data transmission.

In the past, most experiments of chaotic cryptosystem were carried out between two circuit boards in which there was connected by an ideal secure channel. This ideal secure channel did not induce signal-loss. However, for the Internet, packet-loss during data transmission commonly

found. Online real-time multimedia applications are becoming the major services of e-commerce. As the demand of transmitting real-time multimedia data on top of non-guarantee protocol through internet is sharply increasing, it is worthwhile and highly in-need of a chaotic real-time cryptosystem on top of these non-guarantee protocols over Internet for the provision efficient and secure real-time and multi-media applications.

From the algorithmic and implementation point of view, nearly all chaotic cryptosystems being proposed involve long floating point arithmetic calculations. The computational complexity of most chaotic cryptosystems suffers most from the complex floating point arithmetic calculations. It will in turns affect the overall performance and efficiency of the cryptosystems [3][19].

Last but not least, most of the non-linear equations being proposed in the past had narrow key-range with chaotic behavior. It is, therefore, an intrinsic problem to find suitable parameters as the keys for data encryption and decryption.

In order to overcome all these problems, a cryptosystem based on Chaos Theory is proposed such that its chaotic behavior can be used to construct an unpredictable value for mixing with original data for encryption and decryption. This chaotic encryption equation is also used to perform real-time data encryption and decryption for data transmitted using such non-guarantee protocols over Internet.

1.3 Criteria for Designing Chaotic Cryptosystems

When designing chaotic equations for data encryption, it is important to consider the time for data encryption (and decryption) and the level of security. The following were several important criteria for the design of a good chaotic cipher [3][15].

1.3.1 The computation time for encryption and decryption

The computation time for encryption and decryption depends on the complexity of equations and the value of state variable

A) The complexity of equations

The lower the complexity of the equations, the shorter the computation time will be. If the complexity of equation was low, it would obviously reduce the computation time during data encryption and decryption. On the other hand, if the complexity of equation was high, a longer time would be needed for data encryption and decryption.

So in order to choose an equation with lower complexity, a discrete chaotic map is suggested. If the nature of chaotic equation was a discrete map, it would only involve basic arithmetic operations like summations, subtractions, multiplications and divisions etc. On the other hand, if the nature of chaotic equation was a continuous flow, it would involve differential or integration type operations when calculating the value of next state variable.

B) The value(s) of state variable(s)

From the data complexity point of view, an integral value of state variable is more preferable. If the value of state variable was an integer, it would take a shorter time for computing the value of the next state variable. On the other hand, if the value of state variable was a floating point number, it would need a longer time for computing the value of the next state variable.

1.3.2 The level of security

Most chaotic encryption methods are basically symmetric key encryption in which both encryption and decryption key being use the same set of chaotic equations. In most of the case, the parameters of these chaotic equations and their initial values of state variable will be used as the encryption keys (the symmetric keys).

Hence, the level of security will depend on two primitive factors: the key length and the output of encrypted cipher.

A) Key Length and Numbers of Keys

If the key length or numbers of keys are small, it would shorten the time of cryptanalysis of the keys. However, it will impose an intrinsic problem for setting the key length because for most chaotic equations, it would only allow a relatively narrow range of parameter to be chosen with chaotic behavior.

The traditional key value of chaotic equation is floating point number. It means that the key length would be increased based on the precision value of floating point number.

However, as mentioned before, floating point number would substantially increase the computation time. This would also lead to contradiction for designing a good chaotic encryption method as computational complexity and system efficiency is one of the major factors for the design of cryptosystem, especially in a real-time cryptosystem. So in this paper, an integral valued-key is being proposed for the design of real-time cryptosystem.

B) Number of set of chaotic equations

A large number of sets of chaotic equations will induce difficulties in cryptanalysis (and hence a better security level). If the number of set of chaotic equations was small, it would be easier for cryptanalysis.

1.4 Encryption Techniques using Chaotic Equations

In the chaotic encryption world, there are two main ways to perform encryption. One is by Synchronization Technique, the other one is by Self-synchronization Technique.

1.4.1 Synchronization Technique

Chaotic real-time encryption based on Synchronization Technique uses two identical sets of Chaotic Map Equations like Logistic Equation [5][6][13][17].

At the transmitter, the chaotic equation generates chaotic signal. It then uses an add-up function to mix up (mask) the original signal with the chaotic signal.

At the receiver, the chaotic equation generates chaotic signal which is same as that in the transmitter side. It then uses a reverse mix-up (masking) function to retrieve the original signal using the received signal and the generated chaotic signal (Fig. 1).

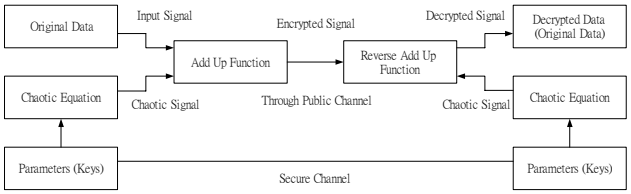

Fig. 1 Architecture of Synchronization Technique in a typical Real-time Chaotic Cryptosystem

Advantage

Data can be transmitted using a non-guarantee protocol by means of these special mechanisms.

The number of packet must also be transmitted with the data packet. Therefore, if a data packet was lost, synchronizing the chaotic signals on both sides would be based on the number of packets.

Disadvantage

Key must be shared in the secure channel.

1.4.2 Self-synchronization Technique

Chaotic real-time encryption using two identical sets of Chaotic Flow Equations such as Lorenz Equation and Rossler Equation [22]. Receiver will receive a driven message from Sender.

At the transmitter, the original signal will be imposed with the chaotic equation and outputs a chaotic signal which will be transmitted to the receiver.

At the receiver, the received chaotic signal will be injected into the chaotic equation and outputs original signal. Under this chaotic real-time encryption scheme, certain time is needed to synchronize those state variables with the transmitter (Fig. 2).

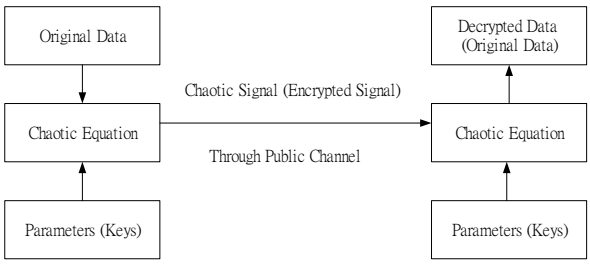

Fig. 2 Architecture of Self-Synchronization Technique in a typical Real-time Chaotic Cryptosystem

From the implementation point of view, this real-time encryption scheme may not be suitable for most online real-time application. The main reason is that it is possible to lose RTP packets. If the receiver misses some RTP packets, self-synchronization cannot be achieved.

Advantage

Key can be shared in the public channel.

Disadvantage

Data must be transmitted in guarantee protocol.

Iteration time is also transmitted with the data packet. If any data packet is lost, it cannot synchronize both sides' state variables. In order to synchronize both sides' state variables, the receiver needs to have the previous state variables.

2. Linear Congruential Generator with Switching Algorithm

2.1 Introduction

A typical Linear Congruential Generator [23] (LCG) is used for finding a suitable parameter to be the key for encryption, which is given by:

$$Xn+1 = (AXn+B) \bmod C \quad (1)$$

Usual Parameters: A = 7141, B=54773, C=259200
Initial Condition: $X_0 = 0$

After analyzing the parameters of the LCG Equation, it is concluded that Parameter B should be the best choice as encryption key for chaotic cryptosystem.

However, only one key is not enough because the key length will be short. The LCG should, therefore, be redesigned. It was easy to increase the number of equations by using different Parameter B. It, however, came to a question that which equations should be used to encrypt.

In this paper, Switching Algorithm was, therefore, taken the role of switching among those equations for encryption [9]. The output of the Switching Algorithm was then used to change Parameter B of LCG Equation regularly. In this experiment, Parameter B was changed regularly per 30 iterations.

The following is the mechanism of the LCG with Switching Algorithm (SLCG):

Equation 1: Xn+1 = (7141Xn+1) mod 259200

Equation 2: Xn+1 = (7141Xn+2) mod 259200

Equation 3: Xn+1 = (7141Xn+3) mod 259200

…

Equation 259200: Xn+1 = (7141Xn+259200) mod 259200

In general, the SLCG is given by:

$$Xn+1 = (7141Xn+Bkey) \bmod 259200 \quad (2)$$

The output of the Switch Algorithm plus 1 will be equal to the selected equation's number.

For instance, for a SLCG with Bkey = 1, Xn = 100, the SLCG will then be given by:-

$$Xn+1 = (7141Xn+1) \bmod 259200 \quad (3)$$

The output of the SLCG will be given by Xn+1 = 195701.

Selected equation's number = 195701 + 1 = 195702

Equation 195702: Xn+1 = (7141Xn+195702) mod 259200 would be selected for encryption and decryption.

Using this method, the Bkey would be, therefore, used as the key for chaotic encryption and decryption.

At the analysis stage, the trajectory distribution of SLCG's chaotic signal was affected by Parameter X. It highly suggested that Parameter X could be a key of chaotic equation.

2.2. Analyzing Parameter B

Firstly, Parameter A, Parameter C and Parameter X were fixed. Parameter B was changed from 0 to 259200 and the Bifurcation diagram was plotted.

SLCG's Bifurcation diagram was plotted for the desired range of parameter B. Fig. 3a depicts the SLCG's Bifurcation diagram for Parameter B when changing from 0 to 259200. It shows SLCG has the chaotic behavior when Parameter B is within 1 and 259200. SLCG's Bifurcation diagram was plotted for a small range of parameter B. From Fig. 3b, it shows that SLCG's Bifurcation diagram for Parameter B when changing from 0 to 10. It shows SLCG has chaotic behavior when Parameter B is within 1 and 10.

2.3 Analyzing State Variable X

Parameter A, Parameter B and Parameter C were fixed. State variable X was changed from 0 to 259200 to draw the Bifurcation diagram. SLCG's Bifurcation diagram was plotted for the desired range of X. From Fig. 4a, it shows SLCG's Bifurcation diagram when X is changing from 0 to 259200. It shows that SLCG has the chaotic behavior when X is within 1 and 259200. SLCG's Bifurcation diagram was plotted for a small range of X. From Fig. 4b, it shows SLCG's Bifurcation diagram for state variable X when changing from 0 to 10. It shows that SLCG has the chaotic behavior when X is within 1 and 10. Figs. 4a and 4b show chaotic behavior for state variable X ranging from 1 to 259200. In other words, state variable X is suitable for being a key for encryption when it is within 1 and 259200.

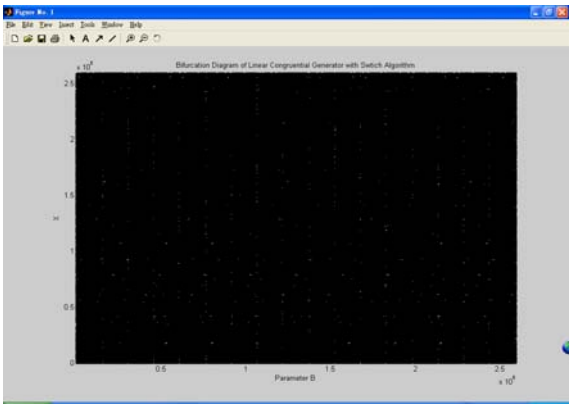

Fig. 3a Bifurcation Diagram for SLCG
(Range of B was from 0 to 259200)

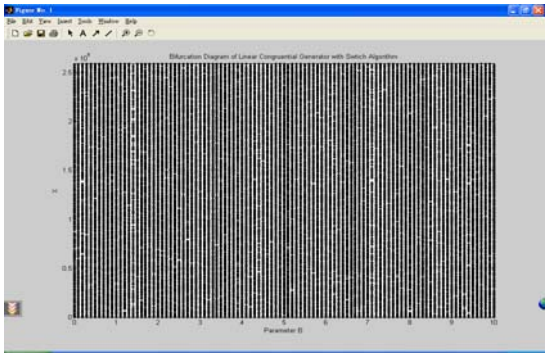

Fig. 3b. Bifurcation Diagram for SLCG
(Range of B was from 0 to 10)

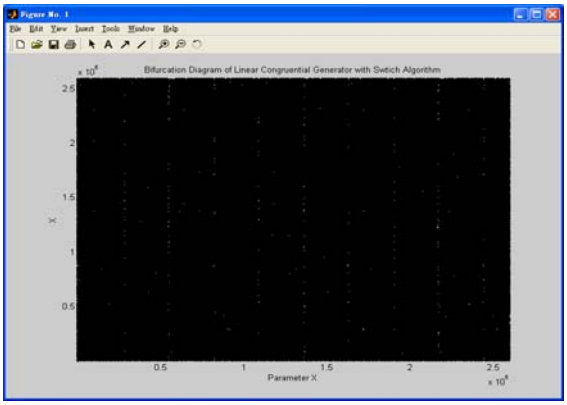

Fig. 4a. Bifurcation Diagram for SLCG
(Range of X was from 0 to 259200)

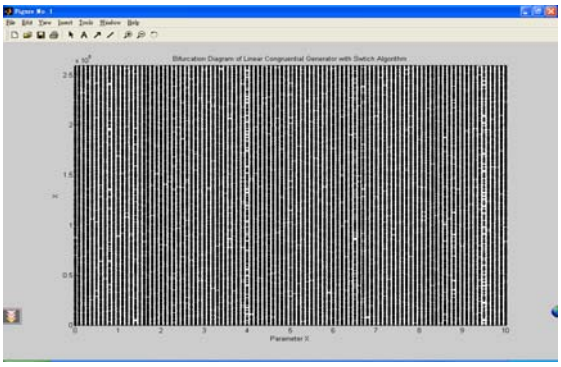

Fig. 4b. Bifurcation Diagram for SLCG
(Range of X was from 0 to 10)

3. Implementation

From the implementation point of view, Java Media Framework (JMF) was used to implement this SLCG-based cryptosystem. Program was used to transfer and receive PCM Audio using RTP packet under the framework of JMF. Each packet was encrypted by chaotic equation before transmission and was decrypted after receiving the data[1].

3.1 System Architecture

Transmitter first located the audio file, target IP Address, port number and specified Packetizer which could encapsulated audio data into RTP packet. During packetizing, LCG Engine encrypted these audio data before packet transmission.

Receiver located the source IP Address, port number and specified Depacketizer which could decapsulated RTP packet back to audio data in a stream of bytes. During depacketizing, LCG Engine decrypted these received packets. Receiver could then play the original audio. Figs. 5 and 6 depict the original system framework of JMF and the system architecture of chaotic cryptosystem with JMF using SLCG.

[1] Sun Microsystems, 1999, JavaTM Media Framework API Guide http://java.sun.com/products/java-media/jmf/2.1.1/guide/index.html

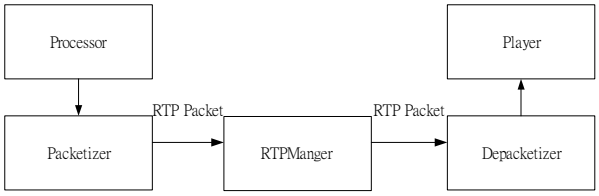

Fig. 5 Original Architecture of JMF

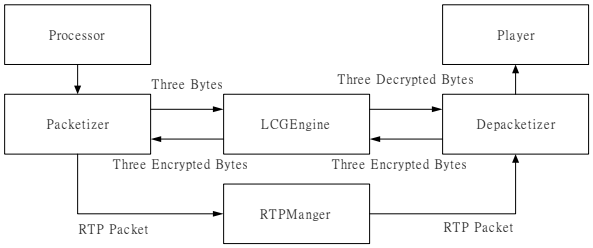

Fig. 6 System architecture of Chaotic Encryption with JMF using SLCG

In the PCM Audio's RTP package, first three bytes were reserved as the number of transferred package in data region of the package. Receiver then synchronized the sequence of chaotic signal with sender based on the number of transferred packet received.

For Example:-

Data size of one PCM audio's RTP Package is 480 bytes

First three bytes were reserved as the number of transferred package and 480-3 bytes were for audio data

If encrypt three bytes each time, we need to perform iterations by (480-3)/3 = 159 times for one package.

When the receiver received the 1st packet, it would generate 159 chaotic signals for the 1st packet.

When receiver lost the 2nd packet and received the 3rd package, it needed to generate 159*2 chaotic signals for the third packet. It was the only way to synchronize the sequence of chaotic signal for the 3rd packet.

When the receiver received the nth package before and then received the mth packet, it needs to generate 159*(m-n) chaotic signals for the mth packet for decryption

4. Results and Analysis

4.1 System efficiency

The transmitter program would transmit a PCM Audio file by RTP packet through the Internet. The time for encryption was calculated at transmitter program. The receiver program would receive RTP packet with audio data through the Internet. The time for decryption was calculated at transmitter program.

In the experiment, time for encryption, time for decryption and total time for encryption and decryption was collected by testing with a PCM audio file of 36 seconds in duration. Audio data was encrypted and decrypted by two different equations. Table 1 summarizes comparison in time for encryption and decryption between SLCG and Logistic Equation.

Table 1 Table of time encryption and decryption between SLCG and Logistic Equation

	SLCG-based Cryptosystem	Logistic-based Cryptosystem	Improvement
Average Time for Encryption	1044.7 ms	1137.9 ms	8.1905%
Average Time of Decryption	220.8 ms	373.6 ms	40.8994%
Average Total Time for Encryption and Decryption	1265.5 ms	1511.5 ms	16.2752%

From Table 1, LCG with Switch Algorithm used shorter total time for Encryption and decryption than Logistic Equation did. LCG with Switch Algorithm saved about 16% on time of average total time for encryption and decryption by comparing with Logistic Equation and it had obviously improvement on the Time for Encryption and decryption. One possible reason may be that integer value was used instead of real number.

4.2 Level of Security

There are two methods for measuring the security level: 1. Distribution of chaotic signal, 2. Key length of chaotic equation

A wider distribution of chaotic signal generates higher security level. If distribution of chaotic signal had narrower distribution, it meant it would have higher frequency on certain values of chaotic signal. Hacker could attack the encrypted signal based on the higher frequency chaotic signal. If the key length of the chaotic equation was large, it would have higher security level. Otherwise, if key length of chaotic equation was small and hacker knew the chaotic equation, hacker could simply try all combination of keys.

4.2.1 Trajectory Distribution Analysis

In trajectory distribution analysis, diagrams of trajectory distribution were collected. Diagrams were plotted by changing parameters of SLCG equation and Logistic equation.

A) SLCG-based Cryptosystem

The diagram of Trajectory Distribution of SLCG Equation with Switch Algorithm was first plotted. The following was the experiment result. It was based on the following Switch Algorithm

$$A * X + 1 = 259200 \quad (4)$$

Firstly, Parameter X was fixed at 1. Parameter B was changed from 1 to 10. The specific range of chaotic signal with higher frequency would remain the same when B is changing as shown in Figs. 7a and 7b.

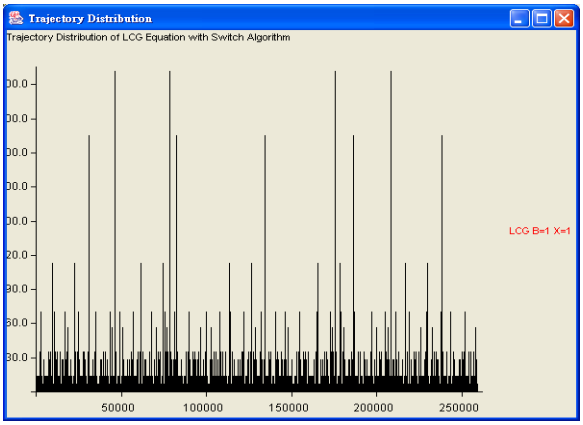

Fig. 7a SLCG Parameter X were fixed at 1 and Parameter B were fixed at 1

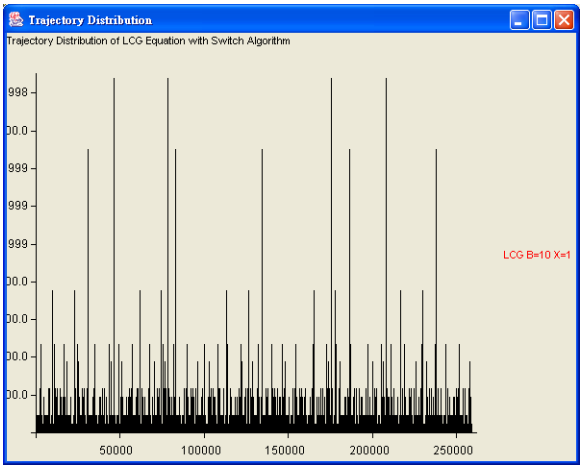

Fig. 7b SLCG Parameter X were fixed at 1 and Parameter B were fixed at 10

Secondly, Parameter B was fixed at 1. Parameter X was changed from 1 to 10. The specific range of chaotic signal with higher frequency would be changed when X was changing. As revealed from Figs. 7c and 7d, some specific range of chaotic signal was of higher frequency. These specific ranges, however, would be changed when the initial condition (Parameter X) was changed. This experiment highly suggested chaotic signals with higher frequency are difficult to be predicted because as the Parameter X changes, the Trajectory Distribution will be changed.

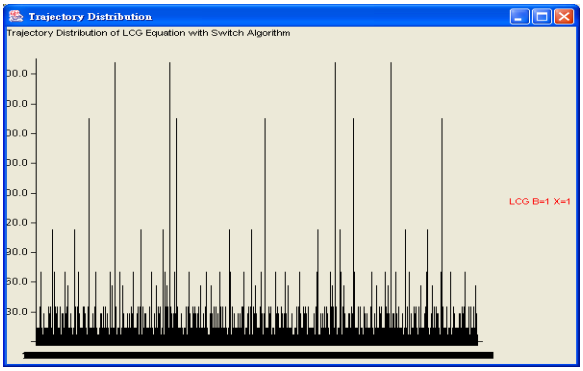

Fig. 7c SLCG Parameter B were fixed at 1 and Parameter X were fixed at 1

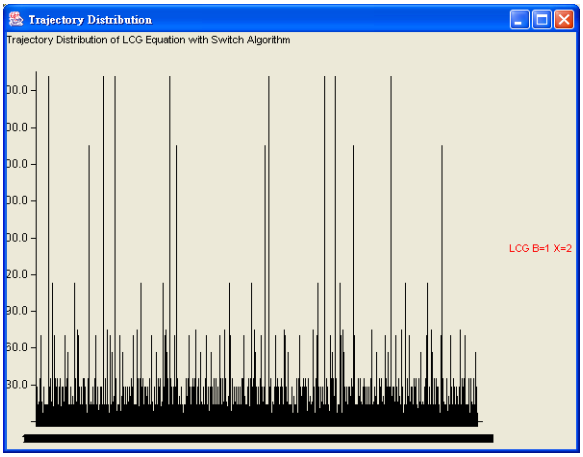

Fig. 7d SLCG Parameter B were fixed at 1 and Parameter X were fixed at 10

B) Logistic-based Cryptosystem

Firstly, Parameter X was fixed at 0.1 and Parameter A was changed from 3.6 to 4

From Figs. 8a – 8c the frequency of the specific range of chaotic signal were not evenly distributed when Parameter A was equal to 3.6 & 3.8.

From Fig. 8c, the frequency of specific range of chaotic signal was evenly distributed when Parameter A was equal to 4.

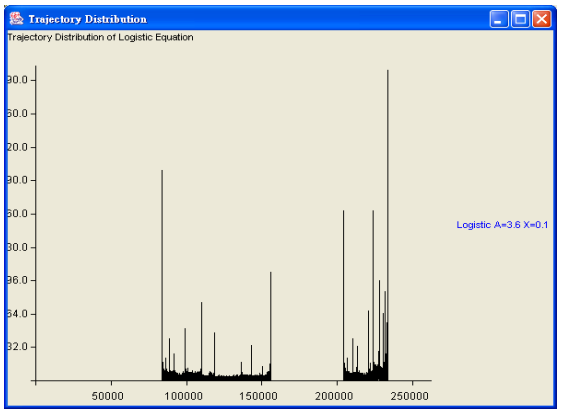

Fig. 8a Logistic Parameter X were fixed at 0.1 and Parameter A were fixed at 3.6

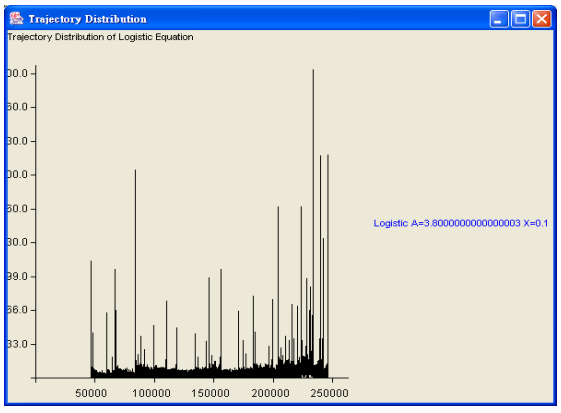

Fig. 8b Logistic Parameter X were fixed at 0.1 and Parameter A were fixed at 3.8

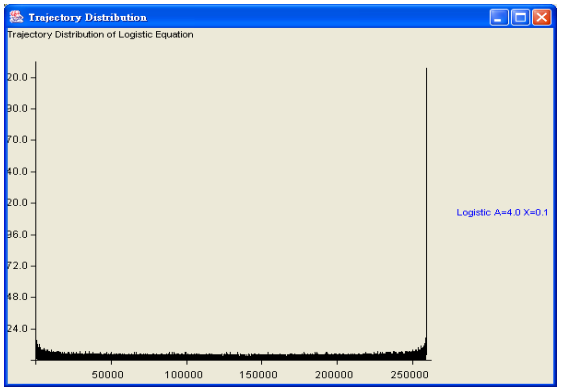

Fig. 8c Logistic Parameter X were fixed at 0.1 and Parameter A were fixed at 4.0

Secondly, Parameter A was fixed at 4 and Parameter X was changed from 0.1 to 0.4

From the Figs. 9a to 9d, the shape of the trajectory distribution of chaotic signal remained the same when X=0.1, 0.2, 0.3, 0.4. Some specific ranges of chaotic signal were of higher frequency. These specific range, however, would be near the lowest and highest chaotic signal even when the initial condition was changed.

Therefore, the chaotic signal is easier to be predicted with higher frequency. That means it would have lower level of security.

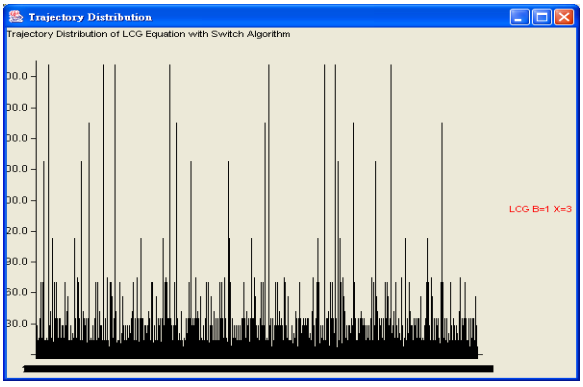

Fig. 9a Logistic Parameter A were fixed at 4.0 and Parameter X were fixed at 0.1

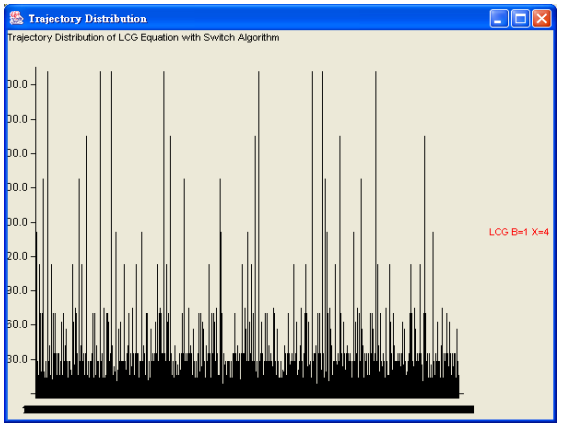

Fig. 9b Logistic Parameter A were fixed at 4.0 and Parameter X were fixed at 0.2

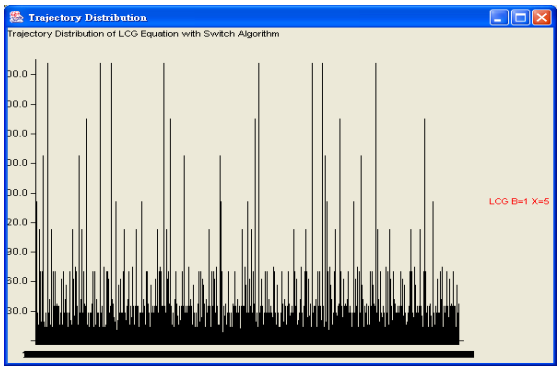

Fig. 9c Logistic Parameter A were fixed at 4.0 and Parameter X were fixed at 0.3

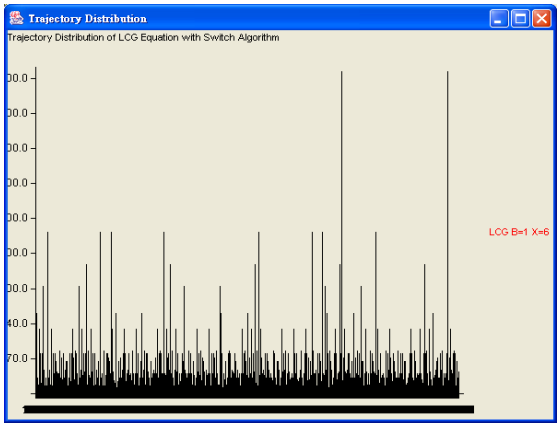

Fig. 9d Logistic Parameter A were fixed at 4.0 and Parameter X were fixed at 0.4

4.2.2 Key Length Analysis

In key length analysis, key length of SLCG and Logistic Equation were calculated as following:

A) SLCG-based Cryptosystem

$$Xn+1 = 7141*Xn+B \bmod 259200 \quad (5)$$

From the design phase, Parameter B is suitable to be a key of chaotic equation.

From the analysis phase, Parameter X is sensitive to the trajectory distribution of chaotic signal. Therefore, it has the potential to be a key of chaotic equation. Parameter B and Parameter X are suitable for being a key from 1 to 259200.

From Switch Algorithm: Xn+1 = 7141*Xn+Bkey mod 259200. Bkey is suitable for being a key from 1 to 259200

Key Length

= Range of B * Range of X * Range of Bkey

= 259200 * 259200 * 259200 = 17414258688000000

=

1111011101111000101101000011110010000000000000000000000 (2)

SLCG needed to use 55-bit to represent key. Although there is no limitation for the number of iterations to trigger Switch Algorithm, it was not suitable to use a very large number of iterations in order to trigger Switch Algorithm. If it is too large, it would not trigger the Switch Algorithm to change the parameter B.

B) Logistic-based Cryptosystem

While for Logistic-based Cryptosystem, which was not suitable to use A as a Key. The trajectory distribution of chaotic signal would be affected by Parameter A. It was suitable to use X as a Key. But Parameter X was only suitable for any real number within 0 and 1 excluding 0 and 1. In Java, it only had 52 bits to represent the floating point number. It could only have 2^53 combinations[2]. Therefore, Logistic had 52-bit key.

From the above calculations, the key length of SLCG Equation is larger than the key length of Logistic Equation. It implies that SLCG Equation generate a higher security level as the time for cryptanalysis of the keys would be increased. Their key length was still too small when compare with SSL 128-bit key length. It was because most of chaotic equation shows chaotic behavior in a small range of parameter. Table 2 shows the comparison of key length between SLCG and Logistic Equation.

Table 2 Table of key length between SLCG and Logistic Equation.

	Calculation of Key Length	Bits of Key Length
SLCG-based Cryptosystem	Range of B * Range of X * Range of Bkey = 259200 * 259200 * 259200 = 17414258688000000	55-bit
Logistic-based Cryptosystem	Any real number within 0 to 1 excluding 0 and 1, Java could use 52 bits to represent the floating point number	52-bit

5. Conclusion

It is still a problem in using self-synchronization over a non-guarantee protocol. It means that keys could not be shared through the public channel. The only solution to this is by synchronization. The keys, however, had to be shared through secure channel. Obviously, if keys could be shared through public channel, a lot of computation cost could be saved as secure channel are not needed to be built.

Switch Algorithm was introduced in this work. By combining LCG equation with Switch Algorithm (SLCG), a chaotic region was found in the Bifurcation diagrams. By analyzing the Bifurcation diagram by changing different parameters, Parameter B and X were found to be suitable for being the keys of encryption. It was because chaotic region was found in the Bifurcation diagrams when changing Parameter B and Parameter X.

SLCG had shorter time for encryption and decryption than Logistic Equation. SLCG showed a greater performance on time for encryption and decryption because it could save 16% of time on encryption and decryption. This result was really important for transmitting real time audio player over the Internet.

SLCG's trajectory distribution of chaotic signal changed when Parameter X was changed. Logistic Equation's trajectory distribution, however, would not change when Parameter X was changed. This highly suggested that SLCG was difficult to be predicted because the trajectory distribution of SLCG changed when Parameter X was changed. Key length of SLCG was 55-bit while key length of Logistic Equation was 52-bit. SLCG had larger key length than Logistic Equation. This implied that SLCG had a higher security level that Logistic Equation. Their key length was still small as compared with SSL 128-bit key length. It was because most of chaotic equations have small range of parameter which shows chaotic behavior.

From the above comparisons, it could be concluded that SLCG would be better to be an encryption algorithm based on time performance and security level than Logistic Equation.

6. Future Works

Although SLCG was better than Logistic-based cryptosystem from the above analysis, key length of SLCG can be increased to enhance the level of security. Like SSL, it already had 128-bit key length.

Current study includes the adoption of a polynomial-based SLCG cryptosystem to further increase the level of security as shown in the following equation.

$$Xn+1=(P1Xn^{(n-1)}+P2Xn^{(n-2)}+\ldots+PiXn^{(n-i)}+\ldots+Pn-1Xn+Pn) \bmod D \quad (6)$$

P1,…,Pn would be the parameters of the equation

Further research has to be done to locate and explore a suitable set of parameters in chaotic region for the implementation of chaotic ciphers.

Acknowledgement

The authors would like to thank the Chaotic Neural Processing (CNP) Research Group and iJADE Development Group of The Hong Kong Polytechnic University for providing support and facilities. This work was partially supported by the CORN project G-T850, iJADE Projects including A-PG50 and Z09H research grants of the Hong Kong Polytechnic University.

References

[1] K. T. Alligood, *Chaos: an introduction to dynamical systems*, Springer, New York, 1996.
[2] E. Alvarez, A. Fernandez, P. Garca, J. Jimenez, A. Marcano, "New approach to chaotic encryption," *Physics Letters A*, vol. 263, 373-375, 1999.
[3] M. S. Baptista, "Cryptography with chaos," *Physics Letters*

[2] Pittsburgh Supercomputing Center (PSC), Carnegie Mellon University, Nov-1999, The IEEE standard for floating point arithmetic, http://www.psc.edu/general/software/packages/ieee/ieee.html

A, vol. 240, pp. 50-54, 1998.
[4] F. Beritelli, E. Di Cola, L. Fortuna, F. Italia, "Multilayer chaotic encryption for secure communications in packet switching networks," *in Proc. Communication Technology 2000 ICCT'2000*, vol. 2, 2000, pp. 1575 -1582.
[5] F. Chiaraluce, L. Ciccarelli, E. Gambi, P. Pierleoni, M. Reginelli, "A New Chaotic Algorithm for Video Encryption Consumer Electronics," *IEEE Trans. on Consumer Electronics*, vol. 48, no. 4, pp. 838 -844, 2002.
[6] K. M. Cuomo, A. V. Oppenheim, S. H. Strogatz, "Synchronization of Lorenz-based chaotic circuits with applications to communications," *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 40, no. 10, pp. 626 -633, 1993.
[7] F. Dachselt, K. Kelber, K., W. Schwarz, "Discrete-time chaotic encryption systems. III. Cryptographical analysis," *IEEE Trans. on Circuits and Systems I: Fundamental Theory and Applications*, vol. 45, no. 9, pp. 983 -988, 1998.
[8] R. L. Devaney, *A First Course in Chaotic Dynamic System*, Addison Wesley, 1992.
[9] J. Q. Fang, Y. Hong, G. Chen, "Switching manifold approach to chaos synchronization," *Physical Review E*, vol. 59, no. 3, pp. 1251-1259, 1993.
[10] M. Gotz, K. Kelber, W. Schwarz, "Discrete-time chaotic encryption systems. I. Statistical design approach," *IEEE Trans. on Circuits and Systems I: Fundamental Theory and Applications*, vol. 44, no. 10, pp. 963 -970, 1997.
[11] J. L. Hudson, O. E., Rossler, H. C. Killory, "Chaos in a four-variable piecewise-linear system of differential equations," *IEEE Trans. on Circuits and Systems*, vol. 35, no. 7, pp. 902 -908, 1988.
[12] T. Kapitaniak, *Controlling chaos: theoretical and practical methods in non- linear dynamics*, Academic Press, London, San Diego, 1996.
[13] G. Kolumban, M. P. Kennedy, L. O. Chua, "The role of synchronization in digital communications using chaos," *IEEE Trans. on Fundamentals of digital communications Circuits and Systems I: Fundamental Theory and Applications*, vol. 44, no. 10, pp. 927-936, 1997.
[14] R. S. T. Lee, F*uzzy-Neuro Approach to Agent Applications - From the AI Perspective to Modern Ontology*, Springer-Verlag, Heidelberg, New York, 2004.
[15] S. Li, *Analyzes and New Designs of Digital Chaotic Cipher*, PhD Dissertation, Xi'an Jiaotong University, 1993.
[16] A. Menezes, P. van Oorschot, and S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997.
[17] M. J. Ogorzalek, "Taming chaos I. Synchronization," *IEEE Trans. on Circuits and Systems I: Fundamental Theory and Applications*, vol. 40, no. 10, pp. 693 -699, 1993.
[18] M. J. Ogorzalek, "Taming chaos II. Control," *IEEE Trans. on Circuits and Systems I: Fundamental Theory and Applications*, vol. 40, no. 10, pp. 700 -706, 1993.
[19] M. I. Sobhy, A. Shehata, "Secure Computer Communication Using Chaotic Algorithm," *Int'l Journal of Bifurcation and Chaos*, vol. 10, no. 12, pp. 997-1000, 2000.
[20] M. I. Sobhy, A. Shehata, "Chaotic algorithms for data encryption," *in Proc. of Acoustics, Speech, and Signal Processing 2001 (ICASSP '01)*, vol. 2, 2001, pp. 997 -1000.
[21] M. I. Sobhy, A. Shehata, "Methods of attacking chaotic encryption and countermeasures," *in Proc. of Acoustics, Speech, and Signal Processing 2001 (ICASSP '01)*, vol. 2, 2001, pp. 1001 -1004.
[22] C. Sparrow, "An introduction to the Lorenz equations," *IEEE Trans. on Circuits and Systems*, vol. 30, no. 8, pp. 533 -542, 1983.
[23] J. C. Sprott, *Chaos and time-series analysis*, Oxford University Press, 2003.

Raymond LEE received his B.Sc. from Hong Kong University in 1989. He received his M. Sc degree in Information Technology and Ph. D. from Hong Kong Polytechnic University in 1997 and 2000 respectively. After graduation from Hong Kong University, he had joined the Hong Kong Government in the Hong Kong Observatory as meteorological scientist on weather forecasting and developing meteorological telecommunication information systems from 1989 to 1993. Prior to joining Hong Kong Polytechnic University in 1998, he had also been worked as MIS Managers and System Consultants in Hong Kong. He is now an Associate Professor in the Department of Computing at the Hong Kong Polytechnic University, working in the areas of Intelligent Agent Technology, neural network and pattern recognition. He is the Founder/Director of iJDG (intelligent Java Development Group) and CNP (Chaotic Neural Processing) group in Hong Kong Polytechnic University. His current research interests include: Artificial Intelligence, Intelligent E-Commerce Systems, Intelligent Agents, Weather Simulation and Forecasting, Chaotic Neural Networks. He is a member of IEEE and ACM.

Henry Lam received his BA Computing from Hong Kong Polytechnic University in 2003. He is now working as the Research Assistant in the CNP (Chaotic Neural Processing) group in the Department of Computing of Hong Kong Polytechnic University. His major research areas include: chaotic cryptosystems, chaotic neural networks.