Comparison of a Chaotic Cryptosystem with Other Cryptography Systems

Ahmed S. Alshammari
Department of Electrical Engineering
College of Engineering, University of Hail
Hail, Saudi Arabia
ahm.alshammari@uoh.edu.sa

Abstract—**The keyspace of a cryptography system must be long enough in order to protect it from brute force attacks. The One-Time Pad (OTP) encryption is unconditionally secure because of its truly random keystream that is used only once. This paper proposes a new chaotic symmetric cryptosystem approach, comparable to OTP. The proposed system utilizes two Lorenz generators, a main and an auxiliary, where the aim of the second one is to make one of the main Lorenz generator's parameters to vary continually with time in a chaotic manner. This technique was built on digitizing two Lorenz chaotic models to increase the security level. The scrambling scheme was developed and the Lorenz stream cipher binary stream successfully passed the NIST randomness test. The cryptosystem showed a high degree of security, as it had a keyspace of 2^{576}, and it was compared with existing symmetric key cryptography systems, such as DES, 3DES, AES, Blowfish, and OTP.**

Keywords-chaotic; AES; one-time pad; keyspace; Lorenz system; NIST; DES

I. Introduction

Cryptography and cryptanalysis are two primary techniques for facilitating secure communications. Cryptography is used in building secure systems to prevent transmitted data (plaintext, the key, or both) from being intercepted by unauthorized people. Such a system is called a cryptosystem, and cryptanalysis is the process to evaluate it. Furthermore, cryptanalysis is used to recover the data transmitted by detecting weaknesses in a cryptosystem. Cryptanalysis is an important step for evaluating the components of a new cryptosystem, such as its security and reliability [1-3]. Two types of cryptosystems exist: symmetric and asymmetric. A symmetric key cryptosystem uses the same private key in both the transmitter and the receiver to quickly encrypt and decrypt a plaintext, which is suitable for applications requiring high data rates, such as video encryption. Furthermore, symmetric key cryptography is divided into two types: block cipher and stream cipher. The block cipher always encrypts and decrypts the plaintext in the same way by using a fixed binary key. The block cipher is widely used in applications, such as triple Data Encryption Standard (3DES), Advanced Encryption Standard (AES), and Rivest Cipher (RC5) [4-5]. On the other hand, a stream cipher is generated using a random binary stream as a secret key to encrypt plaintext to output known as a keystream. The stream cipher key length is based on cryptosystem's features, and it could range from 32 to 256 bits. The most popular stream cipher cryptosystems are Encryption Algorithm (A5/2), RC4, and Software-Optimised Encryption Algorithm (SEAL). A cryptosystem is asymmetric when it uses two different keys for encryption and decryption, where the first key is publicly distributed and the second key is private. Such cryptosystems are primarily used for small amounts of data, such as authentication, secret key agreement, and digital signature, because they are slower. The public key length could range from 1024 to 4096 bits. A widely used public key algorithm is the Rivest-Shamir-Adleman (RSA) [6]. Evaluating security and performance of a communication system is not an easy task. However, many guidelines exist for enhancing a system's robustness and security. These guidelines introduce many cryptographic requirements for building up and analyzing a new chaos-based cryptosystem. The major cryptographic requirements and analysis for a chaos-based cryptography system are discussed in this paper. Furthermore, the system should pass the tests for confusion and diffusion, randomness of bit stream sequence, encryption speed and sensitivity of mismatched key [7-24]. According to [22], the conditions for characterizing an algorithm as safe are:

- The time required for breaking an algorithm is longer than the time the encrypted information must be kept secret.
- The cost required to break the algorithm is greater than the value of the encrypted information.
- The amount of data encrypted using one key is less than the amount of data required to break the algorithm.

Unconditionally secure and computationally secure are two terms used to describe an algorithm. In an unconditionally secured algorithm the cryptanalyst does not have complete information to retrieve the plaintext, no matter how much of the cipher text is present. A computationally secure algorithm is hard to break with the available or even future resources [24]. According to [25] the breaking of an algorithm is categorized as:

- Total break: the cryptanalyst finds the key.
- Global deduction: the cryptanalyst determines a different algorithm that decrypts the cipher text without the key.

Corresponding author: Ahmed S. Alshammari

- Local deduction: the cryptanalyst extracts a plaintext from the received cipher text.
- Information deduction: the cryptanalyst finds some information from the key or the plaintext.

II. THE PROPOSED CRYPTOSYSTEM

Figure 1 displays a block diagram of the proposed cryptosystem. One of the main Lorenz generator's parameter varies with time based on the auxiliary Lorenz output signal. The plaintext is encrypted by the key stream generated from the main Lorenz generator using the multiplication block. The cipher text is generated and transmitted to the receiver. The receiver system has identical main and auxiliary Lorenz generators with the transmitter. The decryption process is started by multiplying the keystream with the cipher text to retrieve the plaintext. The first step is to synchronize the clocks of the transmitter and the receiver. In order to decrypt the cipher text, the receiver and the transmitter must have identical chaotic generators. This means that any intruder must have complete knowledge of the chaotic system's parameters and initial conditions in order to be able to decipher the message.

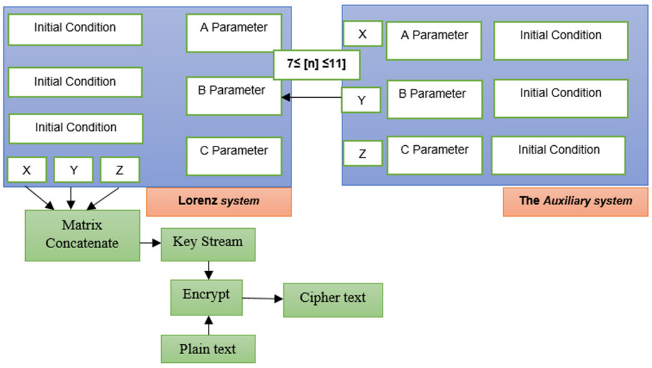

Fig. 1. Block diagram of the cryptosystem.

The encryption technique utilizes the output of the main Lorenz generator to encrypt the data stream, while both the main and the auxiliary Lorenz Generators are based on (1). The Lorenz system is described by the following state equations, written in differential equation form, where A, B, and C are system parameters, and $\dot{x}$, $\dot{y}$, and $\dot{z}$ are state variables.

$$\begin{aligned} \dot{x} &= A(y-x) \\ \dot{y} &= Bx - y - xz \qquad (1) \\ \dot{z} &= xy - Cz \end{aligned}$$

The A parameter of the main system is continuously varied by the auxiliary generator. Furthermore, the parameters and initial conditions of the cryptosystem are changing in every usage to satisfy the third condition of OTP. The constant block was used to manipulate the 32 bit length. Thus, the last 12 bits, starting from the least significant bit, of the x-state keystream were extracted. The variable selector block is used to extract a subject of rows from each matrix. The same operation is used for the y-state keystream. However, the 20 bits are extracted from the y-state of 32 bits, starting from the least significant bit. After that, 12 bits and 20 bits are concatenated using the matrix concatenation block to produce 32 bits. Then, the 32 bits are serialized using the buffered block. The auxiliary Lorenz generator was pre-configured with a different set of initial conditions and system parameters. This system continuously varies the A parameter of the main Lorenz generator. Intensive care was taken to ensure that the main generator always remains in the chaotic region, and the output of the auxiliary Lorenz generator ($A[n]$) remains within the range ($7\leq x[n]\leq 11$). Therefore, the signal response of the main Lorenz generator changes continually in a chaotic manner, based on the parameter supplied by the auxiliary. Figure 2 shows the results from SIMULINK of the Lorenz state variables, x, y, and z.

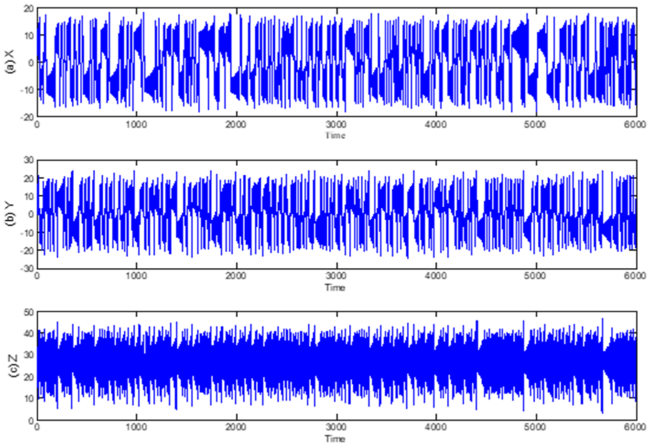

Fig. 2. Lorenz state variables: (a) x-state variable, (b) y-state variable, and (c) z-state variable.

III. RANDOMNESS TEST OF THE KEYSTREAM

The keystream must satisfy the randomness test to avoid any weaknesses in system's security. In this experiment, 100 binary sequences each sized 1,000,000 bits were generated by the Lorenz Generator. The cryptosystem was evaluated using a NIST randomness test, and the chaotic keystream of the proposed generator passed all NIST 800-22 statistical randomness tests .The results are shown in Table I.

TABLE I. NIST RANDOMNESS TEST

Statistical Test	Status	P-value
Frequency	Pass	0.242425
Block Frequency	Pass	0.8221413
Cusum-Forward	Pass	0.252325
Cusum-Reverse	Pass	0.232325
Runs	Pass	0.645146
Long Runs of Ones	Pass	0.344309
Rank	Pass	0.463485
FFT Test	Pass	0.2342325
Non-overlapping	Pass	0.841218
Overlapping	Pass	0.211425
Approximate Entropy	Pass	0.811318
Serial	Pass	0.622146
Linear Complexity	Pass	0.460485

Since the Lorenz system generates an analog signal, an Analog-to-Digital Converter (ADC) is necessary for digital applications. Figure 3 shows the simulation results of the ADC.

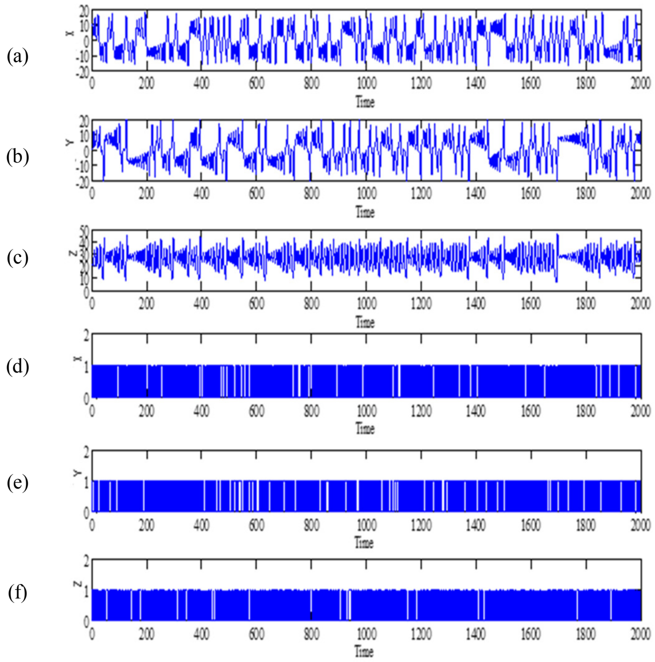

Fig. 3. Analog chaotic signal converted to digital signal. (a) Analog signal of the *x*-state variable, (b) analog signal of the *y*-state variable, (c) analog signal of the *z*-state variable, (d) digital signal of the *x*-state variable, (e) digital signal of the *y*-state variable, and (f) digital signal of the *z*-state variable.

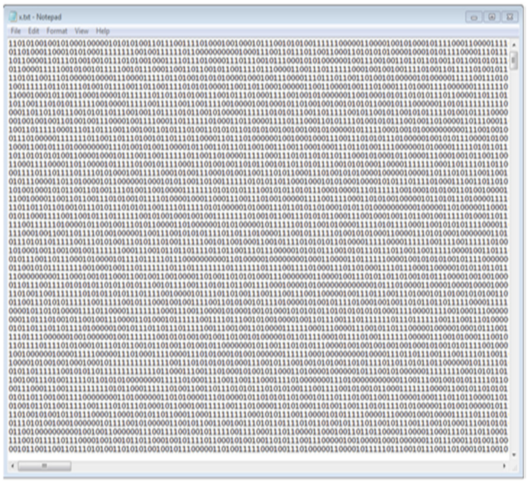

Fig. 4. The keystream of the chaotic cryptosystem.

IV. KEYSPACE OF THE PROPOSED CRYPTOSYSTEM

The transmitter has two Lorenz generators, and each of them has three constants, three initial conditions, and three frequency multipliers. Thus, the total number of the parameters is 18. The word length is 32-bits.The key space of the system is $2^{(18*32)}=2^{576}$. As suggested in [2], the keyspace of a secure cryptosystem should be greater than 2^{100}. Thus, the cryptosystem's keyspace is enough to resist any brute force attack. Table II shows the properties' comparison of the proposed chaotic and five other cryptosystems.

TABLE II. CRYPTOGRAPHY SYSTEMS' PROPERTIES

Factors	Proposed	DES	3DES	Blowfish	AES	OTP
Key Length	576 bits	56 bits	168 bits	Varies between 32 bits to 448 bits	128,192 or 256 bits	Same as Length of the Plaintext (LP)
Cipher Type	Symmetric stream cipher	Symmetric block cipher	Symmetric block cipher	Symmetric block cipher	Symmetric block cipher	Symmetric stream cipher
Block Size	32 bits	64 bits	64 bits	64 bits	128,192 ,or 256	-
Key Space	2^{576}	2^{56}	2^{168}	2^{32}~2^{448}	2^{128}, 2^{192} or 2^{256}	2^{LP}
Security	Considered secure	Not secure against brute force attack	Not Secure	Considered secure	Considered secure	Considered secure

V. CONCLUSION

This paper presented a new cryptosystem based on Lorenz chaotic generators. The system used a stream cipher and the encryption key varied continuously. Furthermore, one Lorenz generator parameter was controlled by an auxiliary chaotic generator for increasing security. This technique was built on digitizing two Lorenz chaotic models, increasing security. The scrambling scheme was developed and Lorenz stream cipher binary stream successfully passed the NIST randomness test. Data encryption used a symmetric cipher with a 576-bit key, and system's keyspace was 2^{576}. The proposed cryptosystem was compared with some existing symmetric cryptography systems such as DES, 3DES, AES, blowfish and OTP in terms of key length and keyspace. Security analysis showed that the system has a high degree of security compared to the other communication systems. The proposed approach is comparable to OTP.

REFERENCES

1. A. S. Alshammari, "Synchronization of Two Chaotic Stream Ciphers in Secure CDMA Communication Systems," *Engineering, Technology & Applied Science Research*, vol. 10, no. 4, 2020, pp. 5947-5952, Aug. 2020.
2. I. Ahmad, A. Saaban, A. Ibrahin, and M. Shahzad, "A Research on the Synchronization of Two Novel Chaotic Systems Based on a Nonlinear Active Control Algorithm," *Engineering, Technology & Applied Science Research*, vol. 5, no. 1, pp. 739–747, Feb. 2015.
3. A. Elsharkawi, R. M. El-Sagheer, H. Akah, and H. Taha, "A Novel Image Stream Cipher Based On Dynamic Substitution," *Engineering, Technology & Applied Science Research*, vol. 6, no. 5, pp. 1195–1199, Oct. 2016.
4. A. A. Khare, P. B. Shukla, and S. C. Silakari, "Secure and Fast Chaos based Encryption System using Digital Logic Circuit," *International Journal of Computer Network and Information Security*, vol. 6, no. 6, pp. 25–33, May 2014.
5. R. Clayton and M. Bond, "Experience Using a Low-Cost FPGA Design to Crack DES Keys," in *Cryptographic Hardware and Embedded*

Systems - CHES 2002, Berlin, Heidelberg, 2003, pp. 579–592, doi: 10.1007/3-540-36400-5_42.

6. T. Kean and A. Duncan, "DES key breaking, encryption and decryption on the XC6216," in *Proceedings. IEEE Symposium on FPGAs for Custom Computing Machines (Cat. No.98TB100251)*, Apr. 1998, pp. 310–311, doi: 10.1109/FPGA.1998.707930.
7. G. Heidari-Bateni and C. D. McGillem, "Chaotic sequences for spread spectrum: an alternative to PN-sequences," in *1992 IEEE International Conference on Selected Topics in Wireless Communications*, Jun. 1992, pp. 437–440, doi: 10.1109/ICWC.1992.200803.
8. X. Wang, X. Wang, J. Zhao, and Z. Zhang, "Chaotic encryption algorithm based on alternant of stream cipher and block cipher," *Nonlinear Dynamics*, vol. 63, no. 4, pp. 587–597, Mar. 2011, doi: 10.1007/s11071-010-9821-4.
9. W. Wong, L. Lee, and K. Wong, "A modified chaotic cryptographic method," *Computer Physics Communications*, vol. 138, no. 3, pp. 234–236, Aug. 2001, doi: 10.1016/S0010-4655(01)00220-X.
10. M. S. Baptista, "Cryptography with chaos," *Physics Letters A*, vol. 240, no. 1, pp. 50–54, Mar. 1998, doi: 10.1016/S0375-9601(98)00086-3.
11. M. A. Aseeri, M. I. Sobhy, and P. Lee, "Lorenz chaotic model using Filed Programmable Gate Array (FPGA)," in *The 2002 45th Midwest Symposium on Circuits and Systems, 2002. MWSCAS-2002*, Aug. 2002, vol. 1, p. I–527, doi: 10.1109/MWSCAS.2002.1187274.
12. G. R. Goslin, "Guide to using field programmable gate arrays (FPGAs) for application-specific digital signal processing performance," in *High-Speed Computing, Digital Signal Processing, and Filtering Using Reconfigurable Logic*, Oct. 1996, vol. 2914, pp. 321–331, doi: 10.1117/12.255830.
13. K. M. Cuomo, A. V. Oppenheim, and S. H. Strogatz, "Synchronization of Lorenz-based chaotic circuits with applications to communications," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 40, no. 10, pp. 626–633, Oct. 1993, doi: 10.1109/82.246163.
14. L. Cong and W. Xiaofu, "Design and realization of an FPGA-based generator for chaotic frequency hopping sequences," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 48, no. 5, pp. 521–532, May 2001, doi: 10.1109/81.922455.
15. M. S. Azzaz, C. Tanougast, S. Sadoudi, and A. Dandache, "Real-time FPGA implementation of Lorenz's chaotic generator for ciphering telecommunications," in *2009 Joint IEEE North-East Workshop on Circuits and Systems and TAISA Conference*, Jun. 2009, pp. 1–4, doi: 10.1109/NEWCAS.2009.5290495.
16. D. Majumdar, R. Moritz, H. Leung, and J. M. Brent, "An enhanced data rate chaos-based multilevel transceiver design exploiting ergodicity," in *MILCOM 2010 Military Communications Conference*, Oct. 2010, pp. 1256–1261, doi: 10.1109/MILCOM.2010.5680115.
17. P. Giard, G. Kaddoum, F. Gagnon, and C. Thibeault, "FPGA implementation and evaluation of discrete-time chaotic generators circuits," in *IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society*, Oct. 2012, pp. 3221–3224, doi: 10.1109/IECON.2012.6389382.
18. L. Merah, A. Ali-Pacha, N. H. Said, and M. Mamat, "Design and FPGA implementation of Lorenz chaotic system for information security issues," *Applied Mathematical Sciences*, vol. 7, pp. 237–246, 2013, doi: 10.12988/ams.2013.13022.
19. S. Liu, J. Sun, Z. Xu, and Z. Cai, "An Improved Chaos-Based Stream Cipher Algorithm and its VLSI Implementation," in *2008 Fourth International Conference on Networked Computing and Advanced Information Management*, Sep. 2008, vol. 2, pp. 191–197, doi: 10.1109/NCM.2008.11.
20. S. Sadoudi, C. Tanougast, and M. S. Azzaz, "A new robust additive hyperchaos masking algorithm for secure digital communications," in *2013 International Conference on Control, Decision and Information Technologies (CoDIT)*, May 2013, pp. 501–504, doi: 10.1109/CoDIT.2013.6689595.
21. Y. Wu, Y. Zhou, and L. Bao, "Discrete Wheel-Switching Chaotic System and Applications," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, no. 12, pp. 3469–3477, Dec. 2014, doi: 10.1109/TCSI.2014.2336512.
22. 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.
23. G. Álvarez, F. Montoya, M. Romera, and G. Pastor, "Cryptanalyzing a discrete-time chaos synchronization secure communication system," *Chaos, Solitons & Fractals*, vol. 21, no. 3, pp. 689–694, Jul. 2004, doi: 10.1016/j.chaos.2003.12.013.
24. A. Shehata, "Secure Computer Communications and Databases Using Chaotic Encryption Systems," Ph.D. dissertation, University of Kent, 2000.
25. L. R. Knudsen, "Block ciphers-analysis, design and applications", Ph. D. dissertation, Department of Computer Science, Aarhus University, 1994.