SECURED TRANSMISSION USING HYPER-CHAOTIC SYSTEMS IN DISCRETE TIME

Florin Zoreanu DRACMAN, Virgil DOBROTA, Member IEEE
Technical University of Cluj-Napoca, Department of Communications, 26-28 Baritiu Street, 400027
Cluj-Napoca, Romania, Phones: +40-264-401816, Fax: +40-264-597083, Web: http://www.com.utcluj.ro,
E-mail: d_florin@yahoo.com, Virgil.Dobrota@com.utcluj.ro

ABSTRACT

This paper presents a new method of secured transmission by using the hyper-chaotic systems in
discrete time. The chaotic and hyper-chaotic systems have complex behavior, the chaotic signals being random,
but still possible to be anticipated, being depending on the initials conditions. The chaotic trajectories seem to
vary randomly, in spite the fact that they are generated by a deterministic system.

I. Introduction

We have studied the hyper-chaotic systems and the possibility to synchronize two
systems. By using these systems in the ECS laboratory research we develop an algorithm for
encryption and decryption, fulfilling the requirements of modern digital networks. It is based
on a new method called “inclusion method”. In this case the information is included into one
of the system equation and it becomes a new state variable. The decryption consists on the
development of an observer which allows the reconstruction of the confidential message
starting from the only information transmitted to the receiver.

The chaotic cryptography studies based on the inclusion method are under progress
and it is currently hard to evaluate the security level. Anyway the first tests prove a high level
of reliability and encryption speed, according to the required transfer rates in nowadays
modern telecommunications networks. Another advantage is related to rapid convergence of
the two systems in the sense that a 4-byte unencrypted word is reconstructed after 3
successive secured transmissions.

II. Burgers Map for Hyperchaotic-Cryptography

We have a two-dimensional discrete-time hyperchaotic system named “Burgers Map”:

\[
\begin{align*}
    x_i^+ &= (1 - a) \cdot x_i - x_i^2 \\
    x_i^- &= (1 + b) \cdot x_i + x_i \cdot x_j
\end{align*}
\]  

where:
\[
\begin{align*}
    x_i^- &= x_i(k - 1) \\
    x_i^+ &= x_i(k + 1)
\end{align*}
\]
In the next paragraph we compute the Lyapunov exponents for this system.

III. Lyapunov Exponents

For the system (1) we compute the fixed points $f(x)=x$. We have three fixed points:

\[
\begin{align*}
(x_i^*, x_j^*) &= (0,0) \\
(x_i''^*, x_j''^*) &= (-b, \sqrt{a \cdot b}) \\
(x_i'''^*, x_j'''^*) &= (-b, -\sqrt{a \cdot b})
\end{align*}
\]

We consider $a=2.28$ and $b=0.548$. The system (1) can be represented in the generic form: $x^+ = f(x, p)$ with $x = (x_i, x_j)^T \in \mathbb{R}^2$ represents the state vector evaluated at the moment $k$ ($x^+ = x(k+1)$), and $p$ represents the vector of the system parameters.

We implemented a Matlab simulation to calculate the Lyapunov exponents $\lambda_i(x_j)_{i=1,2,3}$ (where $x_j$ represents the fixed points of the system (1)). We compute the Lyapunov exponents according with the following formula:

\[
\lambda_i = \lim_{N \to \infty} \left( \frac{1}{N} \log |q_i(f^N(x_j, p))| \right)
\]

where $q_i$ represents the eigenvalues of the Jacobian matrix evaluated at the stationary points $x_j$.

We given below the simulation results for $N=10^4$:

- for $x_1 = (0, 0)$ we have the Lyapunov exponents vector: 
  \[
  \lambda(x_1) = (0.246885, 0.437007)
  \]
- for $x_2 = (-0.548, 1.117783)$ we have the Lyapunov exponents vector: 
  \[
  \lambda(x_2) = (0.0989761, 0.0989761)
  \]
- for $x_2 = (-0.548, -1.117783)$ we have the Lyapunov exponents vector: 
  \[
  \lambda(x_2) = (0.0989761, 0.0989761)
  \]

Because all the exponents are positive the system (1) is hyperchaotic.
IV. Inclusion Method

In this case we have a cryptographic system with secret key; we can observe the emitter, the transmission line and the receiver.

This algorithm is based on a message injection because the message is not only simply added to the output of the chaotic system, it enters and modify the successive iterations thus modulating or driving the trajectory followed by the chaotic system. This process can be considered as system with double level of precision because the message is deeply masked in the chaotic system. The confidential message is included in one of the system variables at the transmitter. On the transmission line we have only single information which will allow at the receiver the reconstruction of the original message. The transmission line can be an Internet network. At the receiver we have the same chaos generator with the exception that the message is not included. The message is not recover at the receiver, it is reconstituted. We will say that an equations system which allows the reconstitution of the message is an observer. After the synchronization of two systems at the transmitter and the receiver the message is prepare to be decrypted. At the transmitter we have the next system:

\[
\begin{align*}
x_1^* &= (1 - x_3) \cdot x_1 - x_2^3 + x_4 \\
x_2^* &= (1 + x_5) \cdot x_2 + x_1 \cdot x_2 \\
x_3^* &= x_3 \\
x_4^* &= u + mesaj \\
x_5^* &= x_5
\end{align*}
\]

where \( x_1 \) and \( x_5 \) are the encryption keys, \( x_4 \) contains the message, \( x_2 \) represents the encrypted output that can be sent to a remote receiver.

- the initial conditions \( x_1^0 = -0.66 \); \( x_2^0 = 1.05 \)
- the keys \( x_3 = 2.28 \), \( x_4 = 0.548 \)
- \( u=0.09 \) represents a constant.

At the receiver we have the following system:

\[
\begin{align*}
\hat{x}_1^* &= (1 - x_3) \cdot \hat{x}_1 - \hat{x}_2^3 + \hat{x}_4 \\
\hat{x}_2^* &= (1 + x_5) \cdot \hat{x}_2 + \hat{x}_1 \cdot \hat{x}_2 \\
\hat{x}_3^* &= x_3 \\
\hat{x}_4^* &= u \\
\hat{x}_5^* &= x_5
\end{align*}
\]
- $\hat{x}$ = unknown variable
- $\tilde{x}$ = estimate variable

The decipher technique consists in the development of an observer which allow the reconstitution of the message using the only information received from the transmitter. The calculation of the iterative errors between the states of the transmitter and the states of the chaotic generator from the receiver allow the application of some corrections for the reconstitution of the initial message and to accelerate the convergence between the two hyperchaotic systems.

At the transmitter we have the next signal generated by the system:

![Transmitter Signal](image)

Number of iterations

Figure 3. The transmitter signal

The state phase at the receiver:

![State Phase at Receiver](image)

Figure 4. The state phase at the receiver

V. The Delayed Constructor Design

We design the receiver as decision and control block able to reconstruct the data in clear which we call “step by step delayed constructor”. We are able to reconstruct the
message in three steps with three delays. The step by step delayed constructor consists on constructing step by step the transmitter dynamics with some delays, such that each constructed dynamic at the $k$th-iteration arises in the construction of the next dynamic at the $(k-1)$th-iteration until the last one which contains the information at the $(k-3)$th-iteration.

The receiver gets only the transmitter output $x_2$ in this case. We can compute the error $e_j$ between the state estimated by the transmitter $x^i_1$ and the state reproduced by the receiver ($\hat{x}_2$ i.e. $x_2 - \hat{x}_2$)). Now we can compute $e_j$ between $x^i_1$ and $\hat{x}^i_1$ ($x^i_1 - \hat{x}^i_1$):

$$
e^i_1 = x^i_1 - \hat{x}^i_1, \quad (7)$$

$$e^i_2 = x_2 \cdot e^i_1.$$

Then we will find:

$$e^i_j = \frac{e^i_2}{x^i_2} \quad \forall x_2 \neq 0 \quad (8)$$

If $x_2 = 0$ we have a singularity and we must eliminate this singularity. Now we can reconstitute the first state of the receiver depending on the estimate error $\hat{x}^i_1$:

$$\tilde{x}^i_j = \hat{x}^i_j + \frac{e^i_2 \cdot x^i_2}{(x^i_2)^2 + \varepsilon} + e^i_j \cdot (1 - \frac{(x^i_2)^2}{(x^i_2)^2 + \varepsilon}) \quad (9)$$

**The $x_1$ correction**

By correction we understand the implementation of the state $\tilde{x}^i_1$ to compute the receiver state $\bar{x}^i_1$:

$$\bar{x}^i_1 = (1 - x^i_2) \cdot \tilde{x}^i_1 - (x^i_2)^2 + \hat{x}^i_1 \quad (10)$$

**The message reconstruction**

Now we can reconstitute the message by a simple differentiation with three delays.

$$\tilde{e}_j = \hat{x}^i_j - \hat{x}^i_j$$

$$\tilde{e}_j = \hat{x}^i_j - \hat{x}^i_j$$

$$\tilde{e}_i = (message)^j \quad (11)$$

To verify this method we have developed a program using Visual C6. Here it’s an example of the original text

“BIBLIOGRAPHY
Version of 18Sep92

and a cipher text:
VI. Conclusion

Based on simulations we have observed the rapidity to crypt and decrypt the information. Furthermore we have obtained a ratio of 8 to 1 between the cipher text and the original text. The applications to be envisaged are the following: secured transmission in digital networks, videoconference, chat, television etc.

References