Message-embedding from a control-theoretical point of view

Gilles Millérioux, Jose Maria Amigo, Jamal Daafouz

Message-embedding from a control-theoretical point of view

Gilles Millérioux[1], José María Amigó[2], and Jamal Daafouz[3]

[1] Université Henri Poincaré. Centre de Recherche en Automatique de Nancy
ESSTIN, 2 Rue Jean Lamour, Vandœuvre-Les-Nancy, France
[2] Centro de Investigación Operativa, Universidad Miguel Hernández
Avda. de la Universidad s/n. 03202 Elche (Alicante), Spain
[3] Institut National Polytechnique de Lorraine
Centre de Recherche en Automatique de Nancy, Nancy, France.
Email: gilles.millerioux@esstin.uhp-nancy.fr, jm.amigo@umh.es,
jamal.daffouz@ensem.inpl-nancy.fr

Abstract. Many encryption methods involving chaotic dynamics have been proposed in the literature since the early 90's. Most of them mask the confidential information being transmitted through an insecure channel, with a chaotic analog or digital sequence. The recovering of the original information usually calls for reproducing at the receiver side the same chaotic signal as at the transmitter side. The synchronization mechanism of the two chaotic signals is known as *chaos synchronization*. In this communication, a connection between chaotic and conventional encryption is established via the control-theoretical condition of *flatness*, with special emphasis on one of the most attractive schemes, namely, *message-embedding*. The main conclusion can be stated as follows: a message-embedded cryptosystem is equivalent to a conventional self-synchronizing stream cipher under the flatness condition. It follows that both architectures have the same level of security.

1 Introduction

There are basically two classes of chaotic cryptosystems. The first one amounts to numerically computing a great number of iterations of a discrete chaotic system, using the message as initial data (see [12][24] and references therein). This is basically also the strategy in [25][2], where periodic approximations of chaotic automorphisms are used to define substitutions (so-called S-boxes) resistant to linear and differential cryptanalysis. The second class, on which we shall actually focus in this paper, amounts to scrambling the message with a chaotic dynamic.

Various cryptosystems, corresponding to distinct ways of masking a message, have drawn the attention of the researchers over the last years. The most important schemes obeying such a principle are additive masking, chaotic switching, discrete or continuous parameter modulation, two-channel transmission, and message-embedding. Additive masking was first suggested in [6] and [26]. Chaotic switching is also referred to as chaotic modulation or chaos shift keying. Such

a technique has been mostly proposed in the digital communications context. A description with deep insights can be found in [17]. Basically, two kinds of parameter modulations can be distinguished: the discrete [23][8] and the continuous one [11][14][7][4]. The message-embedded technique is given different names in the literature: embedding [18][21], non autonomous modulation [28] or direct chaotic modulation [13]. For a general review on chaotic ciphers and their security, see [1].

In this note we establish a parallelism between some digital chaotic ciphers and conventional stream ciphers. (See [3] for an account on the concept of digital chaotic cryptography and implementations.) In doing so, we will restrict our attention to the message-embedding scheme. We will show that, under the control-theoretical condition of flatness on the transmitter, the chaotic message-embedding ciphers and the conventional self-synchronizing stream ciphers are formally analogue and, therefore, they have the same level of security.

2 Message-embedding

In message-embedding the information m_k is directly injected (or, as it is also usually said, embedded) at the transmitter side in a chaotic dynamic f_θ with states x_k. The resulting system turns into a non-autonomous one since the information acts as an exogenous input. Injecting m_k into the dynamic can be considered as a "modulation" of the phase space. Only a function of m_k and x_k, called the "output" and denoted by y_k, is conveyed through the public channel. The output y_k is usually low dimensional and should be unidimensional in the ideal case. In what follows, we will assume that y_k is a scalar (dimension 1), the transmitter being thus a so-called Single Input Single Output (SISO) system. The nonlinear function describing the chaotic dynamics as well as the output function are both parametrized by a vector θ which is intended to act as the secret key.

We consider two classes of messsage-embedding. The first one corresponds to systems governed by the state equations

$$\begin{cases} x_{k+1} = f_\theta(x_k, m_k) \\ y_k = h_\theta(x_k, m_k) \end{cases}, \tag{1}$$

while the second class corresponds to

$$\begin{cases} x_{k+1} = f_\theta(x_k, m_k) \\ y_k = h'_\theta(x_k) \end{cases}. \tag{2}$$

The systems (1) and (2) differ from each other by their *relative degree*.

Definition 1. *([16] p.139) The relative degree of a system with respect to the quantity m_k is the required number r of iterations of the output y_k so as y_{k+r} depends on m_k which actually appears explicitly in the expression of y_{k+r}.*

Remark 1. For Single Input Single Output (SISO) linear systems, the relative degree r corresponds to the difference between the degree of the denominator and the degree of the numerator in their transfer function.

Based on Definition 1, the relative degree of the system (1) is clearly $r = 0$.

On the other hand, systems (2) have a relative degree r strictly greater than 0. If we assume that r is finite and constant (no time-varying), after iterating r times the state vector x_k the output y_{k+r} reads

$$y_{k+r} = h'_\theta(f^r_\theta(x_k, m_k)) \tag{3}$$

where

$$\begin{aligned} f^i_\theta(x_k, m_k) &= x_k \quad when\ i = 0 \\ &= f_\theta(f^{i-1}_\theta(x_k, m_k), m_{k+i-1}) \quad \forall i \geq 1. \end{aligned}$$

and where m_k appears explicitly, that is, for a given x_k, there exists $m'_k \neq m_k$ such that $y_{k+r} = h'_\theta(f^r_\theta(x_k, m_k)) \neq h'_\theta(f^r_\theta(x_k, m'_k))$ whereas for all $m'_k \neq m_k$, $y_{k+r'} = h'_\theta(f^{r'}_\theta(x_k, m_k)) = h'_\theta(f^{r'}_\theta(x_k, m'_k))$ if $r' < r$.

Two mechanisms have been proposed in the literature to recover m_k: the inverse system approach [9] and the unknown input observer approach [15][5][21] [20][22]. The transmitter exhibits an output behavior that depends both on the internal chaotic state vector x_k and on the input signal m_k. The role of the receiver is to reproduce the input m_k given the only available data y_k (and possibly their iterates). Hence, it really acts as an inverse system. A main problem arising in the inverse approach lies in that the inverse system is likely to have bad performance properties in a noisy context. In such a case, this drawback must be redressed and a refinement of the design is needed. This leads naturally to some schemes called Unknown Input Observers (UIO).

The generic equations governing an inverse system or a UIO for (1) (or (2)) are

$$\begin{cases} \hat{x}_{k+r+1} = \tilde{f}_\theta(\hat{x}_{k+r}, y_k, \ldots, y_{k+r}) \\ \hat{m}_{k+r} = g(\hat{x}_{k+r}, y_k, \ldots, y_{k+r}) \end{cases}, \tag{4}$$

with g such that

$$\hat{m}_{k+r} = g(\hat{x}_{k+r}, y_k, \ldots, y_{k+r}) = m_k \quad \text{when} \quad \hat{x}_{k+r} = x_k. \tag{5}$$

A delay equal to the relative degree r must be introduced for causality sake.

The existence of an inverse system or an UIO is guaranteed under the assumption that the system (1) (or (2)) is left invertible. Looking into left invertibility is out of the scope of the present communication, thus we shall assume hereafter that these conditions are fulfilled.

The functions $\tilde{f}_\theta$ and g must be chosen so as the so-called *synchronization with unknown input* can be ensured, that is

$$\forall \hat{x}_0 \in U \text{ and } \forall m_k, \lim_{k\to\infty} \|x_k - \hat{x}_{k+r}\| = 0 \tag{6}$$

or

$$\exists k_f < \infty : \forall \hat{x}_0 \in U, \forall m_k \text{ and } \forall k \geq k_f,\ \|x_k - \hat{x}_{k+r}\| = 0 \tag{7}$$

where U is a non empty set of initial conditions. (6) corresponds to an asymptotic synchronization with unknown input, while (7) corresponds to a finite time synchronization with unknown input.

Message-embedding is very attractive insofar as the synchronization (6) or (7) can be guaranteed without any restriction on the rate of variation of m_k.

3 Comparative Study

3.1 Stream ciphers

In the case of stream ciphers, the *plaintext* is broken up into blocks of the same length, called symbols and denoted by m_k. A major distinction with respect to the block ciphers lies in that the encryption function e can change for each symbol because it depends on a time-varying key K_k, the *keystream*. Generally, keystreams are generated iteratively by feedback shift registers since they produce pseudo-random sequences in a very efficient way.

There are two classes of stream ciphers: the *synchronous* stream ciphers (SSC) and the *self-synchronizing* stream ciphers (SSSC).

The equations of the transmitter for the SSC are:

$$\begin{cases} K_k = \sigma_\theta^s(K_{k-1}) \\ c_k = e(K_k, m_k) \end{cases} . \tag{8}$$

The keystream K_k is generated by a function σ_θ^s parameterized by θ, the parameter θ acting as the secret *static* (or master) key. The ciphertext c_k is available at the transmitter output and conveyed through the channel.

The transmitter of the SSSC is described by the recursions

$$\begin{cases} K_k = \sigma_\theta^{ss}(c_{k-l}, \ldots, c_{k-l-M}) \\ c_k = e(K_k, m_k) \end{cases} , \tag{9}$$

where σ_θ^{ss} is also a function parameterized by θ that generates the keystream $\{K_k\}$ and l a nonnegative integer. Unlike SSC, K_k does not depend now on an internal dynamic but only on a fixed number of past values of c_k. The quantity M is called the *delay of memorization*. As before, c_k is generated by the encryption function e depending on a time-varying key K_k.

For both SSC and SSSC, the reconstruction of the plaintext requires the synchronization of the two sequences $\{K_k\}$ and $\{\hat{K}_k\}$ produced at the transmitter and the receiver sides, respectively. The inherent determinism allows their synchronization as explained next.

In the SSC case, the decryption is specified at the receiver side by

$$\begin{cases} \hat{K}_k = \sigma_\theta^s(\hat{K}_{k-1}) \\ \hat{m}_k = d(\hat{K}_k, c_k) \end{cases} \tag{10}$$

and, in the SSSC case, by

$$\begin{cases} \hat{K}_k = \sigma_\theta^{ss}(c_{k-l}, \ldots, c_{k-l-M}) \\ \hat{m}_k = d(\hat{K}_k, c_k) \end{cases} . \tag{11}$$

In both cases, the decryption function d is such that

$$\hat{m}_k = d(\hat{K}_k, c_k) = m_k \text{ when } \hat{K}_k = K_k. \tag{12}$$

For the SSC, the keystreams $\{K_k\}$ and $\{\hat{K}_k\}$ result from autonomous recurrences. It turns out that the unique way of achieving the synchronization is to initialize the key of the generators σ_θ^s at both sides at the same value ($\hat{K}_0 = K_0$). Therefore, K_0 is part of the secret static key.

As for the SSSC, θ is the parameter vector of the function σ^{ss}. If the parameters are identical at both sides, the respective keystreams synchronize automatically because σ_θ^{ss} operate, at both sides, on the same quantities, namely the past values of c_k. The ability to self-synchronizing constitutes one of the main advantages of such cryptosystems. Indeed, they are resistant against bit slips on the transmission channel without any additional synchronization flags or interactive protocols for recovering lost synchronization [19].

3.2 Message-embedding and flatness

The results stated in this section are based on the notion of *flatness* (see [10] for an introductory theory)

Definition 2. *(Flatness) A system with dynamic f, input e_k and state vector z_k of dimension n is said to be* flat *if there exists a set of independent variables y_k, referred to as flat outputs, such that all system variables can be expressed as a function of the flat output and a finite number of its backward and/or forward iterates.*

In particular, for Single Input Single Output systems, there exist two functions $\mathcal{F}$ and $\mathcal{G}$ which obey

$$\begin{cases} z_k = \mathcal{F}(y_{k+k_\mathcal{F}}, \ldots, y_{k+k'_\mathcal{F}}) \\ e_k = \mathcal{G}(y_{k+k_\mathcal{G}}, \ldots, y_{k+k'_\mathcal{G}}) \end{cases} . \tag{13}$$

where $k_\mathcal{F}$, $k'_\mathcal{F}$, $k_\mathcal{G}$ and $k'_\mathcal{G}$ are integers.

Remark 2. Regarding the computational aspects, let us mention that the search of the functions $\mathcal{F}$ and $\mathcal{G}$ can be done by elimination techniques [27]. Maxima[4] is a powerful computer algebra software implemented in Lisp, that can be used to tackle this problem.

Proposition 1. *The message-embedding cryptosystem (1) (or (2)) is equivalent to a conventional self-synchronizing stream cipher if the nonlinear dynamic f_θ with output y_k and input m_k is flat.*

[4] available at http://maxima.sourceforge.net

Proof. According to the Definition 2, flatness of (1), with relative degree $r = 0$, means that there exist two functions denoted $\mathcal{F}_\theta^0$ and $\mathcal{G}_\theta^0$ and integers $k_{\mathcal{F}_\theta^0}$, $k'_{\mathcal{F}_\theta^0}$, $k_{\mathcal{G}_\theta^0}$ and $k'_{\mathcal{G}_\theta^0}$ such that

$$\begin{cases} x_k &= \mathcal{F}_\theta^0(y_{k+k_{\mathcal{F}_\theta^0}}, \ldots, y_{k+k'_{\mathcal{F}_\theta^0}}) \\ m_k &= \mathcal{G}_\theta^0(y_{k+k_{\mathcal{G}_\theta^0}}, \ldots, y_{k+k'_{\mathcal{G}_\theta^0}}) \end{cases} . \tag{14}$$

It turns out that (1) is strictly equivalent to

$$\begin{cases} x_k = \mathcal{F}_\theta^0(y_{k+k_{\mathcal{F}_\theta^0}}, \ldots, y_{k+k'_{\mathcal{F}_\theta^0}}) \\ y_k = h_\theta(x_k, m_k) \end{cases} . \tag{15}$$

Comparison of (15) with (9) leads to the following result:

i) The system (1) is equivalent to a self-synchronizing stream cipher (9) with secret static key θ and the correspondences (symbol $\equiv$)

- key generator $\sigma_\theta^{ss} \equiv \mathcal{F}_\theta^0$
- running key $K_k \equiv x_k$
- ciphertext $c_k \equiv y_k$
- encrypting function $e \equiv h_\theta$
- delay of memorization $M \equiv |k_{\mathcal{F}_\theta^0} - k'_{\mathcal{F}_\theta^0}|$.

Besides, according to Definition 2, flatness of (2), with relative degree $r > 0$, means that there also exist two functions denoted $\mathcal{F}_\theta^r$ and $\mathcal{G}_\theta^r$ and integers $k_{\mathcal{F}_\theta^r}$, $k'_{\mathcal{F}_\theta^r}$, $k_{\mathcal{G}_\theta^r}$ and $k'_{\mathcal{G}_\theta^r}$ such that

$$\begin{cases} x_k &= \mathcal{F}_\theta^r(y_{k+k_{\mathcal{F}_\theta^r}}, \ldots, y_{k+k'_{\mathcal{F}_\theta^r}}) \\ m_k &= \mathcal{G}_\theta^r(y_{k+k_{\mathcal{G}_\theta^r}}, \ldots, y_{k+k'_{\mathcal{G}_\theta^r}}) \end{cases} . \tag{16}$$

Taking into account (3), it turns out that (2) is strictly equivalent to

$$\begin{cases} x_k = \mathcal{F}_\theta^r(y_{k+k_{\mathcal{F}_\theta^r}}, \ldots, y_{k+k'_{\mathcal{F}_\theta^r}}) \\ y_{k+r} = h'_\theta(f_\theta^r(x_k, m_k)) \end{cases} , \tag{17}$$

where y_{k+r} depends explicitly on x_k and m_k. Letting $l_{h',f^r}(x_k, m_k) = h'_\theta(f_\theta^r(x_k, m_k))$, identification of (17) with (9) leads to the following result:

ii) The system (2) is equivalent to a self-synchronizing stream cipher (9) with secret static key θ and the correspondences

- key generator $\sigma_\theta^{ss} \equiv \mathcal{F}_\theta^r$
- running key $K_k \equiv x_k$
- ciphertext $c_k \equiv y_k$
- encrypting function $e \equiv l_{h',f^r}$
- delay of memorization $M \equiv |k_{\mathcal{F}_\theta^r} - k'_{\mathcal{F}_\theta^r}|$.

Remark 3. Notice that the set of equations (14) (*resp.* (16)) could be used at the receiver to obtain both x_k and m_k without resorting to a state reconstruction through an inverse system or an Unknown Input Observer like (4). Even more is true: the message m_k can be retrieved in finite time via $\mathcal{G}_\theta^0$ (*resp.* $\mathcal{G}_\theta^r$) and the knowledge of x_k is no longer useful. Indeed, substituting x_k of (14) (resp. (16)) into (5) yields

$$\hat{m}_k = g(\mathcal{F}_\theta^r(y_{k+k_{\mathcal{F}_\theta^r}}, \ldots, y_{k+k'_{\mathcal{F}_\theta^r}}), y_k) = m_k \tag{18}$$

where $r \geq 0$ is the relative degree. However, the computational complexity of $g(\mathcal{F}_\theta^r(y_{k+k_{\mathcal{F}_\theta^r}}, \ldots, y_{k+k'_{\mathcal{F}_\theta^r}}), y_k) \equiv \mathcal{G}_\theta^r(y_{k+k_{\mathcal{G}_\theta^r}}, \ldots, y_{k+k'_{\mathcal{G}_\theta^r}})$ is likely to be high and we are better off if the computation is carried out in a recursive way through a state space approach.

3.3 Example

This simple and academic example illustrates the aforementioned connection between the message-embedding and self-synchronizing cryptosystems. We consider a message-embedded cipher with dynamic f and output function h' of the form

$$\begin{cases} x_{k+1} = Ax_k + B\nu(x_k, m_k) \\ y_k \quad\;\; = Cx_k \end{cases}, \tag{19}$$

where (a) $x_k = (x_k^{(1)}, x_k^{(2)}, x_k^{(3)})$ is a 3-dimensional vector, (b) A, B and C are matrices of adequate dimensions and integer entries ranging between 0 and 255, and (c) all operations are performed modulo 256. Numerically, the matrices read

$$A = \begin{bmatrix} 38 & 1 & 0 \\ 7 & 0 & 1 \\ 4 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix},$$

$$C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}.$$

The function ν is chosen to be a bitwise XOR (denoted $\oplus$) between the components of x_k and the plaintext m_k:

$$\nu(x_k, m_k) = x_k^{(1)} \oplus x_k^{(2)} \oplus x_k^{(3)} \oplus m_k \equiv u_k,$$

where $x_k^{(i)}$ and m_k are meant here to be the corresponding 8-bit representation and u_k the result expressed again as an integer between 0 and 255.

The secret static key is the vector $\theta = (38, 7, 4)$, which is the first column of A. Observe that A is written in companion form.

It can be shown [16] that for systems with the structure (19), the relative degree corresponds to the smallest integer r such that $CA^{r-1}B$ is different from 0. Here, since $CB = 1$, the relative degree of the system is 1. Thus, in accordance with the notation used in the general part, we will write f_θ^1 for the dynamic.

Along the lines mentioned in Remark 2, we obtain the first equation of (16) with $\mathcal{F}_\theta^1$ given as

$$\begin{cases} x_k^{(1)} = y_k \\ x_k^{(2)} = 7y_{k-1} + 4y_{k-2} \\ x_k^{(3)} = 4y_{k-1} \end{cases} \tag{20}$$

and, moreover,

$$u_k = y_{k+1} - 38y_k - 7y_{k-1} - 4y_{k-2}\,. \tag{21}$$

Equations (20) and (21) clearly corroborate that the system is flat. Besides, they provide the values $k_{\mathcal{F}_\theta^1} = 0$, $k'_{\mathcal{F}_\theta^1} = -2$, $k_{\mathcal{G}_\theta^1} = 1$ and $k'_{\mathcal{G}_\theta^1} = -2$. Being the relative degree 1, we must compute y_{k+1}:

$$\begin{aligned} y_{k+1} &= CAx_k + CBu_k \\ &= 38x_k^{(1)} + x_k^{(2)} + x_k^{(1)} \oplus x_k^{(2)} \oplus x_k^{(3)} \oplus m_k \\ &= l_{h',f^1}(x_k, m_k) \end{aligned} \tag{22}$$

From the proof of Proposition 1, part *ii)*, the system (19) is equivalent to a self-synchronizing stream cipher (9) with secret static key $\theta = (38, 7, 4)$ and the correspondences

- key generator $\sigma_\theta^{ss} \equiv \mathcal{F}_\theta^1$ given by Eq. (20)
- running key $K_k \equiv x_k$
- ciphertext $c_k \equiv y_k$
- encryption function $e \equiv l_{h',f^1}$ given by Eq. (22)
- delay of memorization $M \equiv |k_{\mathcal{F}_\theta^1} - k'_{\mathcal{F}_\theta^1}| = 2$.

Retrieving m_k requires to compute (18). Here the function $g(x_k, y_k)$ is an XOR between the components of x_k and $u_k = \nu(x_k, m_k)$, that is, $g(x_k, u_k) = u_k \oplus x_k^{(1)} \oplus x_k^{(2)} \oplus x_k^{(3)}$ where $x_k^{(i)}$ and u_k are meant to be the corresponding 8-bit representation similarly to the function ν. Indeed, $u_k \oplus x_k^{(1)} \oplus x_k^{(2)} \oplus x_k^{(3)} = m_k \oplus x_k^{(1)} \oplus x_k^{(2)} \oplus x_k^{(3)} \oplus x_k^{(1)} \oplus x_k^{(2)} \oplus x_k^{(3)} = m_k$. Being the system flat, $x_k^{(i)}$ can be expressed in terms of delayed outputs as indicated by the function $\mathcal{F}_\theta^1$. Hence, one has

$$m_k = (y_{k+1} - 38y_k - 7y_{k-1} - 4y_{k-2}) \oplus y_k \oplus (7y_{k-1} + 4y_{k-2}) \oplus 4y_{k-1}.$$

4 Conclusion

We have compared the architectures of the chaotic message-embedding cipher and the conventional stream ciphers, thereby establishing a formal parallelism. The main conclusions are the following.

a) If the transmitter of a message-embedding cipher is non-flat, the plaintext is recovered only asymptotically (via an inverse system or an observer). In other words, finite-time synchronization (and hence decryption) is achieved only if the

transmitter implements a flat dynamics. In this case, the resulting chaotic cipher is equivalent to a self-synchronizing stream cipher, the correspondence having been identified in Sect. 3.2.

b) We conclude that both architectures have the same level of security.

References

1. Alvarez, G., Li, S.: Some basic cryptographic requirements for chaos-based cryptosystems. Int. J. Bif. Chaos 16, 2129–2151, 2006.
2. Amigó, J.M., Szczepanski, J., Kocarev, L.: A chaos-based approach to the design of crytographically secure substitutions. *Phys. Lett. A*, 343, 55–60, 2005.
3. Amigó, J.M., Kocarev, L., Szczepanski, J.: Theory and practice of chaotic cryptography. Phys. Lett. A 366, 211–216, 2007.
4. Anstett, F., Millérioux, G., Bloch, G.: Global adaptive synchronization based upon polytopic observers. In *Proc. of IEEE International symposium on circuit and systems, ISCAS'04*, pp. 728–731, Vancouver, Canada, May 2004.
5. Boutayeb, M., Darouach, M., Rafaralahy, H.: Generalized state-space observers for chaotic synchronization and secure communications. *IEEE Trans. Circuits. Syst. I: Fundamental Theo. Appl.* 49, 345–349, 2002.
6. Cuomo, K.M., Oppenheim, A.V., Strogatz, S.H.: Synchronization of lorenz-based chaotic circuits with applications to communications. *IEEE Trans. Circuits. Syst. II: Anal. Digit. Sign. Process* 40, 626–633, 1993.
7. Dedieu, H., Ogorzalek, M.: Identification of chaotic systems based on adaptive synchronization. In *Proc. ECCTD'97*, pp. 290–295, Budapest, 1997.
8. Dedieu, H., Kennedy, M.P., Hasler, M.: Chaos shift keying: modulation and demodulation of a chaotic carrier using self-synchronizing chua's circuits. *IEEE Trans. Circuits. Syst. II: Anal. Digit. Sign. Process* 40, 634–642, 1993.
9. Feldmann, U., Hasler, M., Schwarz W.: Communication by chaotic signals :the inverse system approach. *Int. J. of Circuit Theory Appl.* 24, 551–579, 1996.
10. Fliess, M., Levine, J., Martin, P., Rouchon, P.: Flatness and defect of non-linear systems: introductory theory and examples. *Int. Jour. of Control* 61, 1327–1361, 1995.
11. Fradkov, A.L., Markov, A.Y.: Adaptive synchronization of chaotic systems based on speed-gradient method and passification. *IEEE Trans. Circuits. Syst. I: Fundamental Theo. Appl.* 44, 905–912, 1997.
12. Fridrich, J.: Symmetric ciphers based on two-dimensional chaotic maps. *International Journal of Bifurcation and Chaos* 8, 1259–1284, 1998.
13. Hasler, M.: Synchronization of chaotic systems and transmission of information. *International Journal of Bifurcation and Chaos* 8, 1998.
14. Huijberts H. J. C., Nijmeijer H., Willems, R.: System identification in communication with chaotic systems. *IEEE Trans. Circuits. Syst. I: Fundamental Theo. Appl.* 47, 800–808, 2000.
15. Inoue E., Ushio T.: Chaos communication using unknown input observers. *Electronics and communication in Japan part III: Fundamental Electronic Science* 84, 21–27, 2001.
16. Isidori, A.: *Nonlinear control systems.* Communications and control engineering series. Springer, 1995.

17. Kolumban G., Kennedy M.P., Chua L. O. The role of synchronization in digital communications using chaos - part II: Chaotic modulation and chaotic synchronization. *IEEE Trans. Circuits. Syst. I: Fundamental Theo. Appl.* 45, 1129–1140, 1998.
18. Lian K.Y., Liu, P.: Synchronization with message embedded for generalized lorenz chaotic circuits and its error analysis. *IEEE Trans. Circuits. Syst. I: Fundamental Theo. Appl.* 47, 1418–1424, 2000.
19. Menezes, A.J., Oorschot, P.C., Vanstone, S.A.: Handbook of Applied Cryptography. CRC Prees, 1996.
20. Millérioux, G., Daafouz, J.: An observer-based approach for input independent global chaos synchronization of discrete-time switched systems. *IEEE Trans. Circuits. Syst. I: Fundamental Theo. Appl* 1270–1279, 2003.
21. Millérioux, G., Daafouz, J.: Unknown input observers for message-embedded chaos synchronization of discrete-time systems. *International Journal of Bifurcation and Chaos* 14, 1357–1368, 2004.
22. Millérioux, G., Daafouz, J.: *Chaos in Automatic Control*, chapter Polytopic observers for synchronization of chaotic maps, pp. 323–344. Control Engineering Series. CRC Press, 2006.
23. Parlitz U., Chua L.O., Kocarev L., Halle K.S., Shang, A.: Transmission of digital signals by chaotic synchronization. *International Journal of Bifurcation and Chaos* 3, 973–977, 1993.
24. Schmitz, R.: Use of chaotic dynamical systems in cryptography. *Journal of the Franklin Institute* 338, 429–441, 2001.
25. Szczepanski, J., Amigó, J.M., Michalek, T., Kocarev, L.: Crytographically secure substitutions based on the approximation of mixing maps. *IEEE Trans. Circuits and Systems I : Regular Papers* 52, 443–453, 2005.
26. Wu, C.W., Chua L. O.: A simple way to synchronize chaotic systems with applications to secure communications systems. *International Journal of Bifurcation and Chaos* 3, 1619–1627, 1993.
27. Wang, D.: Elimination theory, methods and practice. *Mathematics and Mathematics-Mechanization*, pp. 91–137, 1991. Available at http://www-calfor.lip6.fr/ wang/.
28. Yang, T.: A survey of chaotic secure communication systems. *Int. J. of Computational Cognition*, 2004. Available at `http://www.YangSky.com/yangijcc.htm`.