Mathematical and Computer Modelling 51 (2010) 239–246

Contents lists available at ScienceDirect

Mathematical and Computer Modelling

journal homepage: www.elsevier.com/locate/mcm

Novel orbit based symmetric cryptosystems

M.N. Vrahatis [a,c,*], G.A. Tsirogiannis [b,c], E.C. Laskari [a,c]

[a] Computational Intelligence Laboratory, Department of Mathematics, University of Patras, GR-26110 Patras, Greece
[b] Department of Engineering Sciences, University of Patras, GR-26110 Patras, Greece
[c] University of Patras Artificial Intelligence Research Center (UPAIRC), University of Patras, GR-26110 Patras, Greece

ARTICLE INFO

Keywords:
Cryptosystems
Symmetric key
Nonlinear mappings
Fixed points
Zeros
Stability

ABSTRACT

During the last few years considerable effort has been devoted to research related to chaotic encryption. In this paper a new symmetric key cryptosystem that exploits the idea of nonlinear mappings and their fixed points to encrypt information is presented. Furthermore, a measure of the quality of the keys used is introduced. The experimental results indicate that the proposed cryptosystem is efficient and secure to ciphertext—only attacks. Finally, three modifications of the basic cryptosystem that render it more robust are presented and efficiency issues are discussed.

1. Introduction

Recently, the complexity of the geometrical signal and the statistical properties of chaotic systems have motivated their application to cryptography, and chaotic encryption has received considerable attention [1–3].

Most of the proposed cryptosystems based on chaos consider the problem of information hiding in a symmetrical scheme. Formally, a *symmetric key cryptosystem* can be defined as follows [4]. Consider an encryption scheme consisting of the sets of encryption and decryption transformations $\{E_e : e \in \mathcal{K}\}$ and $\{D_d : d \in \mathcal{K}\}$, respectively, where $\mathcal{K}$ denotes the key space. The encryption scheme is called symmetric key if for each associated encryption–decryption key pair (e, d) it is computationally "easy" to determine d knowing only e, and to determine e from d. A large variety of classical symmetric key cryptosystems exist [5–8].

In this paper a new symmetric key cryptosystem based on nonlinear dynamical systems is presented. The cryptosystem exploits the idea of nonlinear mappings and their fixed points to encrypt information. In particular, the plaintext (original message) consists of strings of symbols (characters) from a known alphabet (e.g. ASCII), while the ciphertext (encrypted message) consists of fixed points of nonlinear mappings.

Prior to the description of the proposed cryptosystem some background material on nonlinear mappings and their fixed points is required. In general, two-dimensional nonlinear mappings are of the following form:

$$\Phi : \begin{cases} \widehat{x}_1 = \varphi_1(x_1, x_2), \\ \widehat{x}_2 = \varphi_2(x_1, x_2). \end{cases} \tag{1}$$

In a nonlinear mapping of the form (1) there exist points which are invariant, or fixed, under the mapping. These points are commonly referred to as *periodic orbits* of the mapping. A point $X = (x_1, x_2)^\top$ is called a *fixed point* of Φ if $\Phi(X) = X$. It is called a *fixed point of order* p, or *periodic orbit of period* p, if

$$X = \Phi_p^p(X) \equiv \underbrace{\Phi(\Phi \cdots (\Phi(X)) \cdots)}_{p \text{ times}}. \tag{2}$$

[*] Corresponding author at: Computational Intelligence Laboratory, Department of Mathematics, University of Patras, GR-26110 Patras, Greece.
E-mail addresses: vrahatis@math.upatras.gr (M.N. Vrahatis), gtsirog@ceid.upatras.gr (G.A. Tsirogiannis), elena@math.upatras.gr (E.C. Laskari).

0895-7177/$ – see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.mcm.2009.08.011

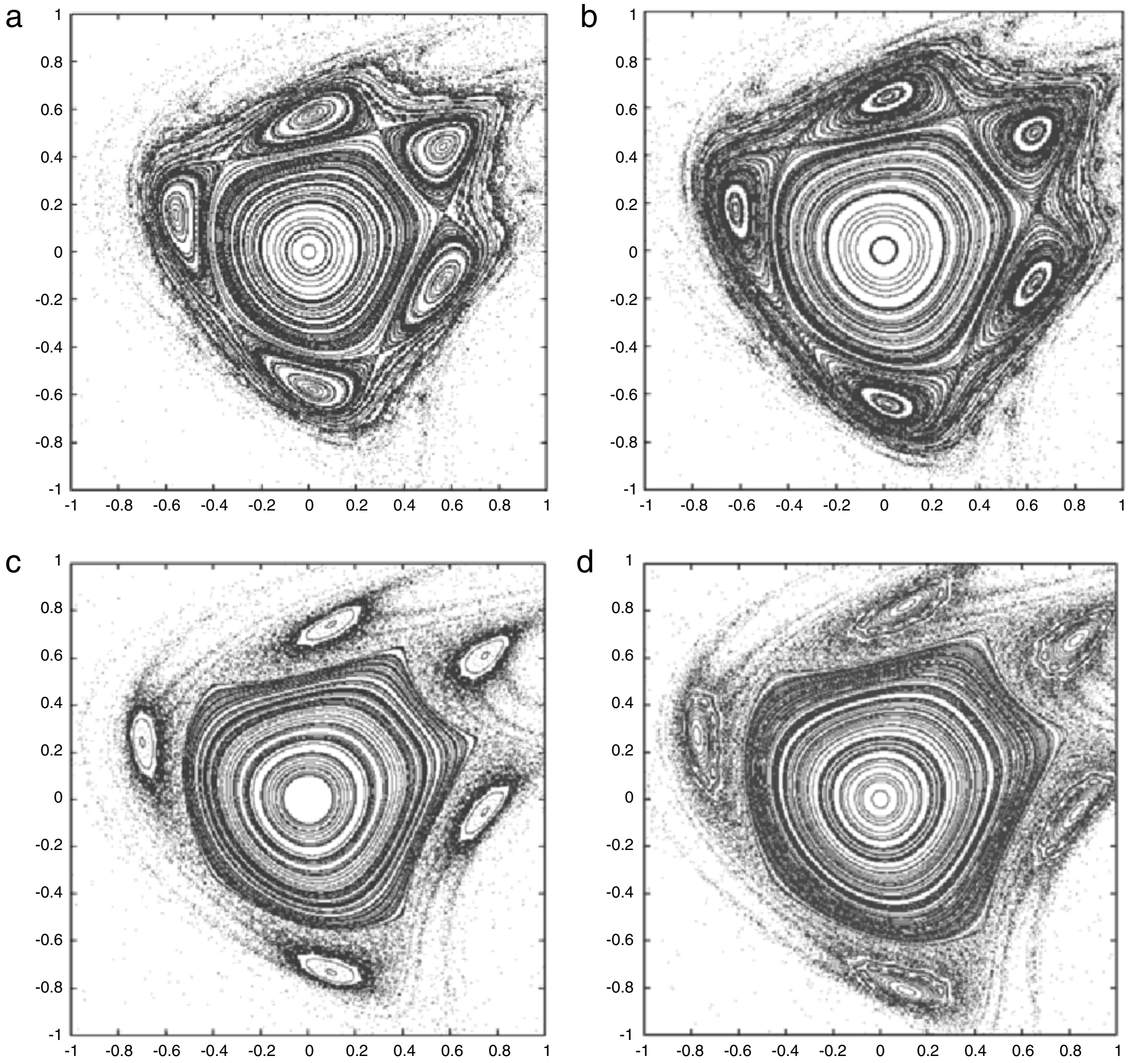

Fig. 1. The Hénon's quadratic area-preserving two-dimensional mapping at the (x_1, x_2) plane for (a) $\cos\alpha = 0.24$, $g(x_1) = -x_1^2$, (b) $\cos\alpha = 0.24$, $g(x_1) = -0.9x_1^2$, (c) $\cos\alpha = 0.2$, $g(x_1) = -x_1^2$, (d) $\cos\alpha = 0.2$, $g(x_1) = -0.9x_1^2$.

A typical example of a nonlinear mapping is Hénon's quadratic area-preserving two-dimensional mapping [9,10]:

$$\Phi : \begin{pmatrix} \widehat{x}_1 \\ \widehat{x}_2 \end{pmatrix} = R(a) \begin{pmatrix} x_1 \\ x_2 + g(x_1) \end{pmatrix}, \tag{3}$$

where $(x_1, x_2)^\top \in \mathbb{R}^2$ and

$$R(a) = \begin{pmatrix} \cos a & -\sin a \\ \sin a & \cos a \end{pmatrix},$$

where $a \in [0, \pi]$ is the rotation angle and $g(x_1) = -x_1^2$. In Fig. 1 instances of Hénon's quadratic area-preserving two-dimensional mapping for different values of its parameters are illustrated.

Although we use the two-dimensional Hénon mapping for illustration, a large variety of mappings can be used, including the Standard mapping [11], the Gingerbreadman mapping [12], the Predator–Prey mapping [13], as well as, higher-dimensional mappings including the Lorenz mapping [14], the Rössler mapping [15] and Hénon's four-dimensional symplectic mapping [16], among others.

For the computation of periodic orbits of an n-dimensional mapping various methods have been proposed. For example, using the CHABIS package [17,18] for Hénon's mapping with $a = \cos^{-1}(0.24)$ and $g(x_1) = -x_1^2$, the following fixed point with period $p = 5$ is computed

$$X_0 = \Phi_5^5(X_0) = (0.5672405470221847, -0.1223202134278941)^\top.$$

To obtain the remaining fixed points of the same order, simple iterations of the mapping on the found fixed point are required,

$$X_1 = \Phi_1^5(X_0) = (0.5672405470221847, 0.4440820516139216)^\top,$$
$$X_2 = \Phi_2^5(X_0) = \Phi_1^5(X_1) = (0.0173925844399303, 0.5800185952239573)^\top,$$
$$X_3 = \Phi_3^5(X_0) = \Phi_1^5(X_2) = (-0.5585984457571741, 0.1560161118011652)^\top,$$
$$X_4 = \Phi_4^5(X_0) = \Phi_1^5(X_3) = (0.0173925844399305, -0.5797160932304572)^\top.$$

Thus, if the mapping and a fixed point with period p are known, all the remaining $(p-1)$ fixed points of the same periodic orbit can be easily computed. The CHABIS package enables us to compute stable and unstable (see Section 3) periodic orbits with periods up to hundreds of thousands [10,16].

To each periodic orbit corresponds a *rotation number* $\sigma = \nu/(2\pi) = i_1/i_2$, where ν is the *frequency of the orbit* and i_1, i_2, are two positive integers. From the sequence in which the above points $X_0, \ldots, X_4$ are created on the (x_1, x_2) plane, we can infer the rotation number of the orbit $\sigma = 1/5$, indicating that it has produced $i_2 = 5$ points, by rotating around the origin $i_1 = 1$ times. For periodic orbits of higher periods of the mapping (3) with the corresponding rotation numbers see [10].

The rest of the paper is organized as follows. In Section 2 the basic form of the proposed cryptosystem is presented. In Section 3 a measure of the quality of the keys used is given. Modifications of the basic form of the cryptosystem, as well as efficiency issues are presented in Section 4. Experimental setup and results for the basic form of the cryptosystem are reported in Section 5. Finally, conclusions are derived in Section 6.

2. Basic form of the proposed cryptosystem

The central idea of the proposed cryptosystem is to hide information in fixed points of an orbit (denoted by O) of a nonlinear mapping. Let X_0, $\Phi_1^p(X_0)$, $\Phi_2^p(X_0)$, $\ldots$, $\Phi_{p-1}^p(X_0)$ be the fixed points of the orbit O with period p. All the fixed points of the orbit O are stored in the order in which they appear by iterating $(p-1)$ times the mapping Φ using the initial fixed point X_0. Such an orbit O can be computed using the Characteristic Bisection method [10,17,18] as well as computational intelligence methods [19,20], or other traditional zero finding methods (such as Newton or Broyden method).

Equivalently, consider the group $G_o = \langle \mathbb{Z}_p, + \rangle$, with elements, $r_i \in \mathbb{Z}_p$, the number of iterations of the mapping Φ over any initial fixed point X_0 of a periodic orbit O with period p. The operation "+" denotes the addition of the number of iterations of the mapping Φ over an initial fixed point X_0 and corresponds to the composition of the mapping Φ, i.e.,

$$\Phi_{r_i}^p(X_0) \circ \Phi_{r_j}^p(X_0) = (\Phi_{r_i}^p \circ \Phi_{r_j}^p)(X_0) = \Phi_{r_i+r_j}^p(X_0).$$

Thus, the addition of the number of iterations of the mapping Φ corresponds to element addition modulo p, i.e., $r_i + r_j \pmod{p}$, where p is the period of the periodic orbit O.

If the nonlinear mapping Φ, the orbit O and a fixed point X_0 of O (which is assumed to be the initial fixed point of O) are known, we can count the number of times that the nonlinear mapping Φ has to be iterated, starting from the initial fixed point X_0, to obtain any other fixed point of the same orbit. It is important to observe that the elements of O, which are placed on a linked-list, have a very specific order which is known only if the nonlinear mapping Φ is known. As will be shown, the encryption–decryption procedures are based on the order of these fixed points.

2.1. Construction of the key

In its basic form the cryptosystem accepts a message m as input (e.g. a sequence of ASCII characters) and produces a matrix $C \in \mathbb{R}^{\text{length}(m)\times 2}$ as output. This implies that every character of the plaintext is encrypted to a fixed point of a specific orbit. Assuming the use of Hénon's mapping as an illustration for the construction of the key, then the key k of the cryptosystem consists of the following:

(a) A nonlinear mapping Φ with the following characteristics: (i) the value of $\alpha \in [0, \pi]$ and (ii) a nonlinear term $g(x_1)$,
(b) a fixed point X_0,
(c) a positive integer p (large enough i.e. $p \geqslant 300$, preferably p should be a prime) which indicates the order of the orbit O to which X_0 belongs.

Note that it is necessary to select periodic orbits whose period exceeds half the cardinality of the character set that is used. For example, if we use the ASCII character set for the message, then $p \geqslant 128$.

The key construction algorithm consists of the following steps.

Key construction algorithm

1. Choose at random a rotation angle $\alpha \in [0, \pi]$.
2. Choose a nonlinear term $g(x_1)$.
3. Choose a quite large positive integer p (preferably p should be a prime).
4. Choose a region of the plane in which the fixed point X_0 must lie.
5. Use a method (e.g. CHABIS [18]) to locate a fixed point X_0.
6. Check if p is the smaller integer that satisfies $X_0 = \Phi_p^p(X_0)$ and terminate.

Note that when p is a prime the sixth step of the construction algorithm can be omitted. Moreover, the choices of Steps 1–4 may not give a fixed point X_0 and in this case a new choice for the input parameters is required.

2.2. Encryption and decryption algorithms

The encryption algorithm takes as input a message m with n characters, $m_1m_2\cdots m_n$ (e.g. in ASCII format), and a key k; and returns a ciphertext $C \in \mathbb{R}^{n\times 2}$. The algorithm is described in the following steps.

Encryption algorithm

```
1. Initialize $C_0 = X_0, i = 1$.
2. Iterate ASCII($m_i$) times the considered nonlinear mapping $\Phi$ starting from the fixed point $C_{i-1}$, to obtain $C_i = \Phi^p_{\mathrm{ASCII}(m_i)}(C_{i-1})$.
3. Proceed to the next character by setting $i = (i + 1)$.
4. **if** $i < n$ **then** goto Step 2, **else** terminate.
```

In reverse, the decryption algorithm takes as an input a ciphertext $C \in \mathbb{R}^{n\times 2}$ and a key k and provides as output the original plaintext m, through the following procedure.

Decryption algorithm

```
1. Initialize $C_0 = X_0, i = 1$.
2. Starting from the fixed point $C_{i-1}$, count the number of iterations, $noi$, of the considered nonlinear mapping $\Phi$ required to satisfy for the first time the relation $C_i = \Phi^p_{noi}(C_{i-1})$. The character $m_i$ of the plaintext is obtained by the equality, $noi = \mathrm{ASCII}(m_i)$.
3. Proceed to the next character by setting $i = (i + 1)$.
4. **if** $i < n$ **then** goto Step 2, **else** terminate.
```

3. Quality of the keys

The essential part of the key for the proposed cryptosystem is the initial fixed point used, since all the remaining fixed points of the same periodic orbit of the nonlinear mapping are generated from it. Notice that the initial fixed point is useless if the mapping is unknown. The initial fixed point that is used for the key can suggest a measure of its quality. This is due to the sensitivity property of the fixed points of a periodic orbit to perturbations. Specifically, the larger the real eigenvalues of the returned Jacobian (see Step 2 of the following stability checking algorithm) calculated at the fixed point, the more sensitive the key to small perturbations. Fixed points that are sensitive to small perturbations comprise periodic orbits that are called *unstable*. In this case, all the fixed points of the unstable periodic orbit can be accurately obtained only if the initial fixed point is known with high accuracy. Any small perturbation to the coordinates of one fixed point leads to large changes to all other fixed points of the same periodic orbit. Thus, even if an adversary gains knowledge of some digits of the initial fixed point of an unstable periodic orbit he cannot generate the remaining fixed points of the specific orbit. Following this property, a definition of *strong keys* is given.

Definition 1. A key of the proposed cryptosystem is called strong, if the initial fixed point used in the key belongs to an unstable periodic orbit of the given nonlinear mapping.

For the verification of the type of a periodic orbit, i.e. stable or unstable, the following stability checking algorithm can be used. The algorithm takes as input the nonlinear n-dimensional mapping Φ and the initial fixed point X_0 that are used, the period p of the periodic orbit to which the point X_0 belongs, and the Jacobian matrix of the mapping Φ.

Stability checking algorithm

```
1. Compute the Jacobian matrix of the mapping $\Phi$ for the initial fixed point $X_0$, and set $J = \mathrm{Jacobian}(X_0)$.
2. **for** $i = 2 : p$ **do**
       $X_i = \Phi^p_1(X_{i-1})$
       $J = \mathrm{Jacobian}(X_i)J$
   **enddo**
3. Compute the eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n \in \mathbb{C}$ of the matrix $J$.
4. **if** $\Im(\lambda_i) \neq 0$ **and** $(\Re(\lambda_i)^2 + \Im(\lambda_i)^2)^{1/2} = 1$, for all $i = 1, 2, \ldots, n$
   **then** the orbit is "STABLE"
   **elseif** $\Im(\lambda_i) \neq 0$ **and** $(\Re(\lambda_i)^2 + \Im(\lambda_i)^2)^{1/2} \neq 1$, for any $i \in \{1, 2, \ldots, n\}$
   **then** the orbit is "COMPLEX UNSTABLE"
   **elseif** $\Im(\lambda_i) = 0$ **and** $\Re(\lambda_i) \neq 1$ for all $i = 1, 2, \ldots, n$
   **then** the orbit is "UNSTABLE"
   **endif**
```

If all the eigenvalues of the Jacobian matrix J are complex and they are located on the unit circle then the orbit is *stable*. Otherwise, if at least one eigenvalue is complex and it is not located on the unit circle then the orbit is *complex unstable*. Finally, if all the eigenvalues are real and their value is not equal to one then the orbit is called *unstable*. In the last case, the larger in magnitude one of the eigenvalues is, the more unstable the orbit is considered to be.

4. Modifications of the basic form of the cryptosystem and efficiency issues

In this section three modified versions of the proposed cryptosystem are presented. These variations aim at making the cryptosystem more robust to possible attacks.

4.1. First modification

Consider the case where an adversary knows two ciphertexts C_i and C_j (i.e. two fixed points that encrypt the characters m_i and m_j of the message respectively) and the number of iterations of the mapping Φ that is required to get C_j from C_i. Then, he can obtain the character m_j of the plaintext. To avoid this a secret initial fixed point for the encryption of any plaintext character can be used and is defined as follows.

Definition 2. A fixed point is called initial fixed point for the encryption of a character m_i if it is used as a starting point to get ASCII(m_i) iterations of the mapping Φ.

Using a secret initial fixed point for the encryption of a character m_i, we have

$$\text{Encryption}(m_i) = \Phi^{p}_{\text{ASCII}(m_i)}(X_{\text{initial}\, m_i}).$$

In the basic form of the cryptosystem it is assumed that the initial fixed point of each character is the fixed point that encrypts the previous character, i.e.,

$$\text{Encryption}(m_i) = \Phi^{p}_{\text{ASCII}(m_i)}\left(\Phi^{p}_{\text{ASCII}(m_{i-1})}(X_0)\right).$$

For the calculation of the initial fixed points of each character of the message, the following formulae are proposed:

$$\begin{cases} X_{\text{initial}\, m_1} = \Phi^{p}_{\beta H(m) (\text{mod } p)}(X_0), \\ X_{\text{initial}\, m_i} = \Phi^{p}_{\gamma \text{ASCII}(m_{i-1}) (\text{mod } p)}(C_{i-1}), \quad \text{for } i = 2, \ldots \end{cases}$$

where β, γ are two positive and large integers which must be appended to the key k, and $H(m)$ is the result of a one-way hash function on the message m (e.g. MD5 [21]). For $\beta = \gamma = 0$, we get the basic form of the proposed cryptosystem. Finally, a random dummy character can be added in front of the message m such that a completely different ciphertext is always obtained.

4.2. Second modification

Consider the case where some information concerning the sorting of the fixed points of the periodic orbit used is revealed to an adversary. In order to confront an attack that is based on frequency analysis of the appearing ciphertexts, in this modification, every plaintext character is encrypted by a fixed point which is independent from the previously used fixed points (that encrypt the previous characters). For the implementation of the described modification, two large positive integers β, γ are chosen and appended to the key k. Moreover, a one-way hash function H that operates over the message m and gives a value $H(m)$, is applied. Then, the product $H(m)\beta$ is computed and employed as a seed to a uniform random number generator. Suppose that each time, i, the number generator is called it returns a random number $\text{rand}(i) \in [0, 1]$. Subsequently to each call we compute the value $\text{rand}_p(i) = \lceil \text{rand}(i) \rceil p$. Thus, the initial fixed point of each character for stream ciphers is given by

$$X_{\text{initial}\, m_i} = \Phi_{\text{rand}_p(i)\gamma (\text{mod } p)}(X_0),$$

and for non-stream ciphers by

$$\begin{cases} X_{\text{initial}\, m_1} = \Phi_{\text{rand}_p(1)\gamma (\text{mod } p)}(X_0), \\ X_{\text{initial}\, m_i} = \Phi_{\text{rand}_p(i)\, \text{ASCII}(m_{i-1})\gamma (\text{mod } p)}(X_0), \quad \text{for } i = 2, \ldots. \end{cases}$$

4.3. Third modification

In this variant the following problem is considered. Given p fixed points of an orbit O in random order, find a mapping Φ that has O as a periodic orbit with period p. This hypothetical case can be tackled by performing a bit-by-bit XOR operation between the fixed point that encrypts a plaintext character with its initial fixed point. Thus, the fixed points of the orbit O are not revealed. Moreover, the ciphertext set becomes of size $\mathit{Card}(\text{ASCII})\, p$ which corresponds to $\mathit{Card}(\text{ASCII})\, p!$ permutations for a brute force attack.

4.4. Efficiency issues

In this section the number of FLOPs required for an encryption or decryption of a single character of the message is provided. For Hénon's mapping and the encryption (or decryption) of the character m_i, using the basic form of the cryptosystem, we have

$$\text{FLOPs}(C(m_i)) = \text{ASCII}(m_i)(6 + \text{FLOPs}(g(x_1))).$$

Table 1
Experimental results for different kinds of plaintexts.

Plaintext type	CPU time for encryption/decryption (s)	Ciphertext percentage in	
		Zeros (%)	Ones (%)
10 000 random characters	≈0.001	49.9999	50.0001
100 000 random characters	≈0.00102	49.9999	50.0001
100 000 char. English text	≈0.00103	49.9999	50.0001
100 000 same characters	≈0.00103	49.9999	50.0001

For the modifications of the encryption/decryption algorithm a small number of FLOPs needs to be added. Specifically, the additional FLOPs are

$$\text{FLOPs}(C(m_i)) \leqslant p(6 + \text{FLOPs}(g(x_1))) + t,$$

where t is a small constant.

5. Experimental setup and results

For the evaluation of the proposed cryptosystem, additionally to the number of FLOPs required which is presented in Section 4.4, the CPU time needed for the encoding (and decoding) of several plaintexts (ciphertexts) is measured. Subsequently, the percentage of zeros and ones contained in the resulting ciphertexts is reported.

For comparative purposes we have tested three different kinds of plaintexts, specifically, common English texts, texts containing random characters and texts that repeat the same character. The results for the last kind of text are important to ensure that even when the plaintext contains some patterns there is no information that can be deduced from the encrypted message. The results obtained from the basic form of the proposed cryptosystem on a P4 machine with 512 MB RAM using Matlab 6.5 are given in Table 1.

Regarding the CPU time required for the encryption/decryption of the texts only a few milliseconds where needed for all kinds of plaintexts, which was expected since only some iterations of the nonlinear mapping are performed (note that the implementation is not optimized for high performance). Furthermore, the only CPU time spent on the computation of the key is due to the computation of the initial fixed point, as the choice of the nonlinear mapping takes no CPU time. Employing a proper method, for example CHABIS [10], fixed points with period of some thousands can be computed in less than a second of CPU time, using a typical modern personal computer.

Regarding the total number of zeros and ones on the resulting ciphertexts the percentages were about 50% 0's and 50% 1's for all types of plaintexts, i.e. random text, English text and repeated characters. This is an important result for the security of the cryptosystem to ciphertext—only attacks as the ciphertexts do not contain information for possible patterns in the corresponding plaintexts.

In Fig. 2(a) and (b), examples of possible keys (nonlinear mapping and initial fixed points) of the proposed cryptosystem are depicted in the (x_1, x_2) plane. Even if an adversary gains knowledge about the employed nonlinear mapping, choosing a slightly different initial fixed point will lead to a different periodic orbit (i.e. ciphertexts). Furthermore, a specific periodic orbit is completely useless when a different mapping (even for a mapping with the same nonlinear term but different rotation angle a) is used. In Fig. 2(c) and (d), the periodic orbit of Fig. 2(a) is placed over two slightly different nonlinear mappings. As shown, these points do not constitute a periodic orbit of neither of the new mappings and, moreover, an adversary cannot find a fixed point of the new mappings without gaining some knowledge about them.

6. Conclusions

In this contribution a new symmetric orbit based cryptosystem is presented. The proposed cryptosystem exploits the idea of nonlinear mappings and their fixed points for encryption. Specifically, the encryption algorithm takes as input a string of characters and outputs a number of fixed points of a nonlinear mapping defined by the key. Furthermore, a measure of the quality of the keys is introduced. This measure is based on the sensitivity of the fixed points of a periodic orbit to perturbations. Thus, by applying the stability checking algorithm one can decide if a key has the desired quality.

The security of the proposed cryptosystem is mainly attributed to the difficulty of sorting the elements of a given periodic orbit when the nonlinear mapping is unknown. Any kind of brute force attack is completely inefficient, since small changes of the parameters of the nonlinear mapping (e.g. rotation angle and nonlinear term), or perturbations of the initial fixed point lead to very different results.

The experimental results indicate that the cryptosystem is quite efficient as it requires a small number of FLOPs for the encryption/decryption of a single character and only some milliseconds of CPU time for the encryption and decryption of messages with 100.000 characters. Moreover, the analysis of ciphertexts obtained by different kinds of plaintexts shows that no pattern information can be deduced from the ciphertexts rendering, thus, the cryptosystem robust to ciphertext-only attacks.

Finally, three different variants of the basic form of the cryptosystem that render it more effective to several types of possible attacks are presented.

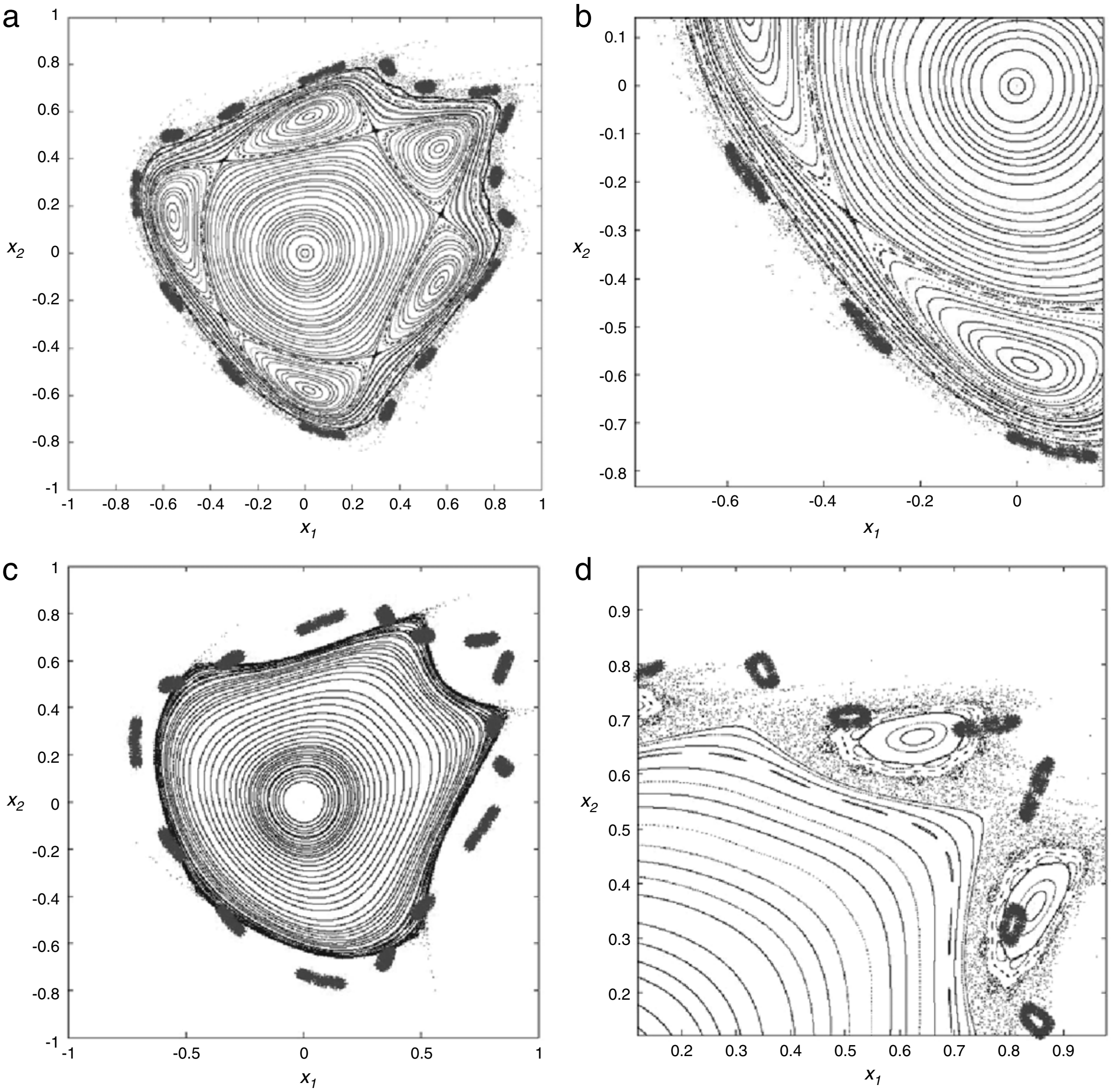

Fig. 2. (a) Fixed points of the Hénon's quadratic area-preserving two-dimensional mapping with $\cos\alpha = 0.24$ and $g(x_1) = -x_1^2$, used as keys of the cryptosystem, (b) details of keys used in (a), (c) keys of (a) placed over the Hénon's quadratic area-preserving two-dimensional mapping with $\cos\alpha = 0.15$ and $g(x_1) = -x_1^2$, (d) keys of (a) placed over the Hénon's quadratic area-preserving two-dimensional mapping with $\cos\alpha = 0.35$ and $g(x_1) = -x_1^2$.

References

[1] T. Yang, C.W. Wu, L.O. Chua, Cryptography based on chaotic systems, IEEE Trans. Circuits Syst. 44 (1997) 469–472.

[2] G. Grassi, S. Mascolo, A system theory approach for designing cryptosystems based on hyperchaos, IEEE Trans. Circuits Syst. 46 (9) (1999) 1135–1138.

[3] F. Dachselt, W. Schwarz, Chaos and cryptography, IEEE Trans. Circuits Syst. 48 (12) (2001) 1498–1509.

[4] A. Menezes, P. Van Oorschot, S. Vanstone, Handbook of Applied Cryptography, CRC Press, 1996.

[5] Data Encryption Standard (DES), US Dept. of Commerce, Dec. 30, 1993, FIPS PUB 46-2 (C13.52).

[6] Advanced Encryption Standard (AES) Fact sheet, October 3, 2000.

[7] X. Lai, On the Design and Security of Block Ciphers, in: ETH Series on Information Processing, Verlag, 1992.

[8] A. Shimizu, S. Miyaguchi, Fast Data Encipherment Algorithm FEAL, IEICE, July 1987.

[9] M. Hénon, Numerical study of quadratic area-preserving mappings, Quart. Appl. Math. 27 (1969) 291–311.

[10] M.N. Vrahatis, An efficient method for locating and computing periodic orbits of nonlinear mappings, J. Comput. Phys. 119 (1995) 105–119.

[11] S.N. Rasband, Chaotic Dynamics of Nonlinear Systems, Wiley, New York, 1990.

[12] R.L. Devaney, A piecewise linear model for the zones of instability of an area preserving map, Physica D 10 (1984) 387–393.

[13] M.J. Smith, Mathematical Ideas in Biology, Cambridge University Press, London, 1968.

[14] E.N. Lorenz, Deterministic nonperiodic flow, J. Atmospheric Sci. 20 (1963) 130–141.

[15] O.E. Rössler, An equation for continuous chaos, Phys. Lett. A 57 (1976) 397–398.

[16] M.N. Vrahatis, H. Isliker, T.C. Bountis, Structure and breakdown of invariant tori in a 4-D mapping model of accelerator dynamics, Internat. J. Bifur. Chaos 7 (1997) 2707–2722.

[17] M.N. Vrahatis, Solving systems of nonlinear equations using the nonzero value of the topological degree, ACM Trans. Math. Software 14 (4) (1988) 312–329.

[18] M.N. Vrahatis, CHABIS: A mathematical software package for locating and evaluating roots of systems of nonlinear equations, ACM Trans. Math. Software 14 (4) (1988) 330–336.

[19] K.E. Parsopoulos, M.N. Vrahatis, Computing periodic orbits of non-differentiable/discontinuous mappings through particle swarm optimization, in: Proc. IEEE 2003 Swarm Intelligence Symposium, 2003, pp. 34–41.

[20] K.E. Parsopoulos, M.N. Vrahatis, On the computation of all global minimizers through particle swarm optimization, IEEE Trans. Evol. Comput. 8 (3) (2004) 211–224.

[21] B. Schneier, One-way hash functions, Dr. Dobb's J. (1991).