Cryptanalyzing a nonlinear chaotic algorithm (NCA) for image encryption

G. Alvarez*[,a], Shujun Li[b]

[a] *Instituto de Física Aplicada, Consejo Superior de Investigaciones Científicas, Serrano 144, 28006–Madrid, Spain*
[b] *Fachbereich Informatik und Informationswissenschaft, Universität Konstanz, Fach M697, Universitätsstraße 10, 78457 Konstanz, Germany*

Abstract

This paper describes the security weakness of a recently proposed image encryption algorithm based on a logistic-like new chaotic map. We show that the chaotic map's distribution is far from ideal, thus making it a bad candidate as a pseudo-random stream generator. As a consequence, the images encrypted with this algorithm are shown to be breakable through different attacks of variable complexity.

1. Introduction

In recent years, there has been an intense research on chaotic cryptography. As a result, a steadily growing number of cryptosystems based on chaos have been proposed during past decades [1, 2, 3, 4, 5, 6, 7]. There are two main approaches to the design of chaotic cryptosystems: analog and digital. This paper is concerned with the latter, digital chaotic ciphers, which are designed for digital computers: in general, one or more chaotic maps are implemented in finite computing precision to encrypt the plain-message in a number of ways [8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]. Also, a number of chaotic ciphers were specially designed for the encryption of digital images and videos [21, Sec. 4.4]. However, many of them are fundamentally flawed by a lack of robustness and security, as showed in [22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39].

In [40], an image encryption algorithm based on a new chaotic map is proposed. Founded on the assumption that "the Logistic map does not satisfy uniform distribution property", the authors designed a new chaotic map called NCA, and defined as:

$$x_{n+1} = (1-\beta^{-4}) \cdot \cot\left(\frac{\alpha}{1+\beta}\right) \cdot \left(1+\frac{1}{\beta}\right)^{\beta} \cdot \tan(\alpha x_n) \cdot (1-x_n)^{\beta}, \tag{1}$$

where $x_n \in (0,1)$, $\alpha \in (0,1.4]$, $\beta \in [5,43]$, or $x_n \in (0,1)$, $\alpha \in (1.4,1.5]$, $\beta \in [9,38]$, or $x_n \in (0,1)$, $\alpha \in (1.5,1.57]$, $\beta \in [3,15]$. We will denote the union of these sets as the key space, $\mathcal{K}$. According to the authors, the ranges of α and β correspond to the chaotic map operating in chaotic regime. This map allegedly presents "good properties of balanced 0-1 ratio, zero co-correlation and ideal nonlinearity".

The orbit generated by this chaotic map for a choice of the parameters α and β in the chaotic region, and starting from an arbitrary initial condition x_0, is used to generate a keystream, k, by following an algorithm

*Corresponding author: Email: gonzalo@iec.csic.es

detailed in [40] and described next. The elements of the keystream are bytes, i.e., integer numbers in the range $0 \leq k_i \leq 255$. Images are encrypted by XOR-ing their pixel information with the keystream, k. The secret key is formed by the operation parameters and the initial point: $K = (\alpha, \beta, x_0)$, where $K \in \mathcal{K}$.

In the next section of this paper, a thorough analysis of the NCA map and a number of attacks on the proposed algorithm are provided. The results show that the NCA map performs very poorly as a tool to generate the keystream for encryption. All computations and image operations were made in Matlab.

2. Attacks on the proposed system

2.1. Statistical analysis

The algorithm presented in [40] corresponds to the category of stream ciphers. They are characterized by the application of simple encryption operations, such as XOR, according to the keystream being used. This keystream can be generated through a random process, or, more often, pseudorandomly from a small secret key, with the intent that it appears random to a computationally bounded attacker.

The keystream generation process of the algorithm under analysis is based on the iteration of the map described by Eq. (1). After 100 iterations starting from a given key, K, every third point of the orbit is used to generate 5 bytes in a real-to-integer conversion process:

> "Divide the first 15 significant digits into five integers with each integer consisting of three digits. For each integer, do mod 256 operation, and another five bytes of data will be generated."

As is well known, an important requirement for the keystream to be statistically random-like is that the underlying chaotic map have a uniform distribution. Some histograms computed for different values of $K \in \mathcal{K}$ are depicted in Fig. 1, corresponding to the central points of the three ranges proposed in [40], and showing that the distribution is far from ideal. As a consequence, such an ill-behaved sequence should never be used as generator of a keystream.

For a number of values of (α, β), we estimated the value of $P(x < 0.001)$, by averaging the corresponding frequencies of ten 100,000-point chaotic orbits with randomly generated values of x_0. With the experimental data, the distribution of $P(x < 0.001)$ with respect to (α, β) is shown in Fig. 2. Considering that the first byte generated from a chaotic state x is $\lfloor 1000x \rfloor$, a large value of $P(x < 0.001)$ means a large value of $P(1000x < 0)$, i.e., a large ratio of zeros in the derived keystream.

To illustrate this point, the distribution of zeros in the keystream as a function of (α, β), has been depicted in Fig. 3. From the observation of the figure, it can be deduced that, at best, about 1.06% of the bytes in the keystream are zero, whereas a well balanced distribution requires that every byte appears in average in 0.39% of the keystream and at random intervals, that is, the total number of zero bytes is 2.7 times larger than it should be. Note that the percentage is much larger than 0.39% for most values of (α, β): when $\alpha \in (0, 1)$ and $\beta \in [5, 43]$, the least percentage is about 11%, and when $\alpha \in (1.4, 1.5], \beta \in [9, 38]$, the least percentage is about 27%. The average of all percentages obtained in our experiments is around 48.8% and 125 times of 0.39%, an extremely large rate. Furthermore, as a result of the method used to transform the real-number orbit points into bytes, most zero bytes appear in bursts.

It is important to note that the choice of x_0 is irrelevant with respect to the distribution of the chaotic orbit.

The statistical analysis in Sec. 4 of [40] was conducted on the encrypted images, whereas it should have been performed besides on the keystream. This analysis would have revealed the inadequacy of Eq. (1) as a pseudorandom generator. The suggested rule 16 from [41] is reproduced here for convenience:

When a keystream cipher is used, the security study should include the statistical test results conducted on the pseudo-random number generator.

Furthermore, it should be noted that even passing all existing statistical tests is a necessary, but not sufficient condition for the security of an algorithm.

In Fig. 4, an image is encrypted using three values from the available key space suggested in [40], which correspond to the best, middle, and poorest performances, respectively. As can be seen, the leaked information in each case corresponds to the percentage of zero bytes shown in Fig. 3. For the latter two

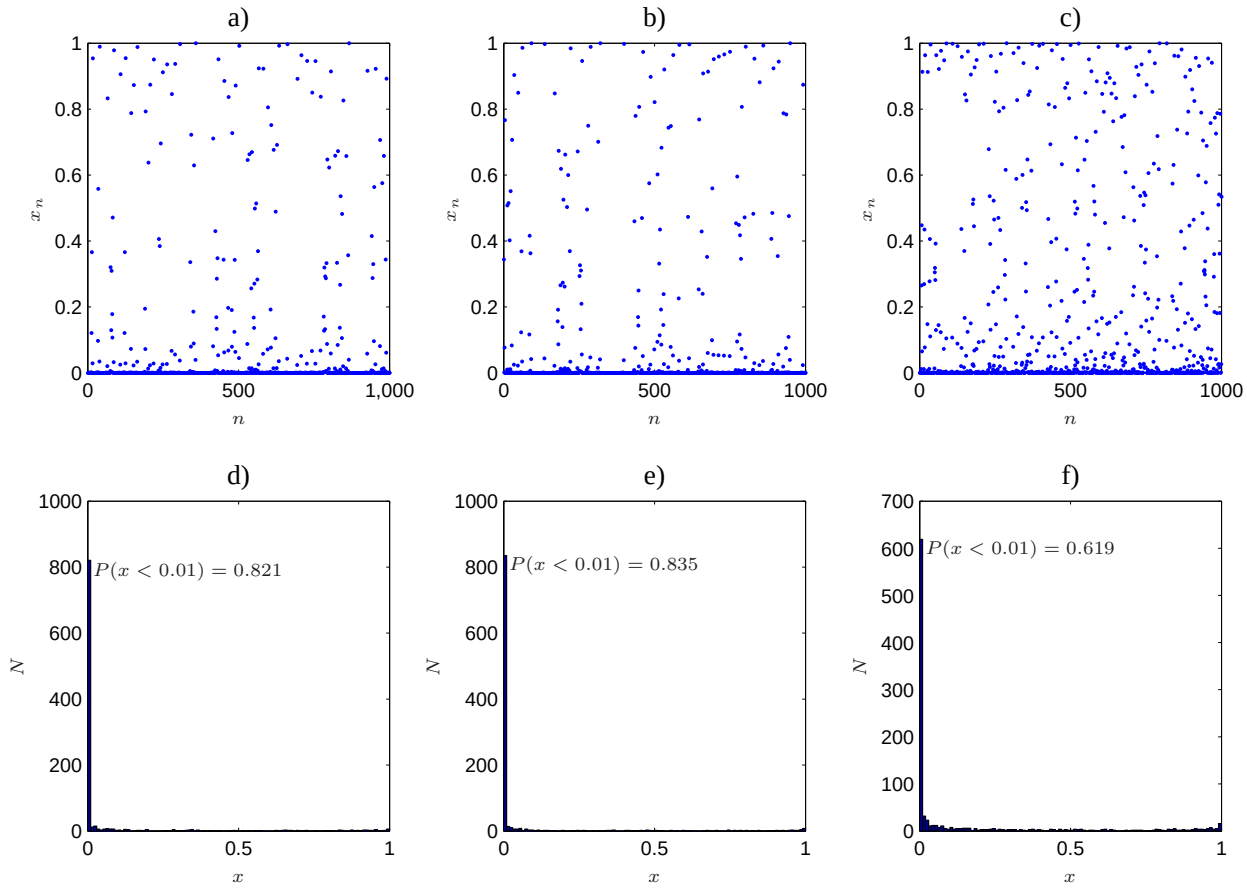


Figure 1: Three different 1000-point orbits. On the top line: orbit points for a) $\alpha = 0.7$, $\beta = 24$; b) $\alpha = 1.45$, $\beta = 23.5$; and c) $\alpha = 1.535$, $\beta = 9$. On the bottom line: d), e), f) 100-bin histograms for the same parameter values, respectively.

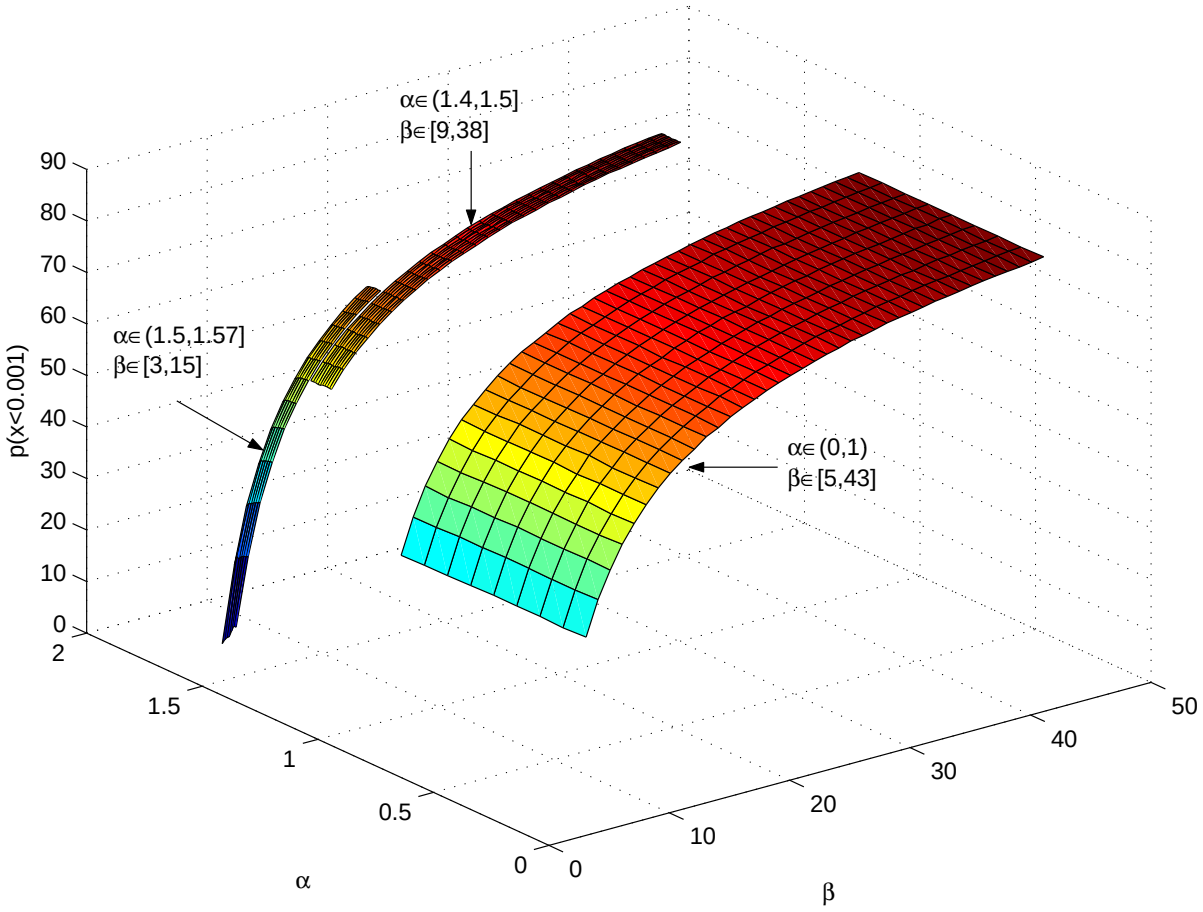


Figure 2: Distribution of $P(x < 0.001)$ as a function of (α, β).

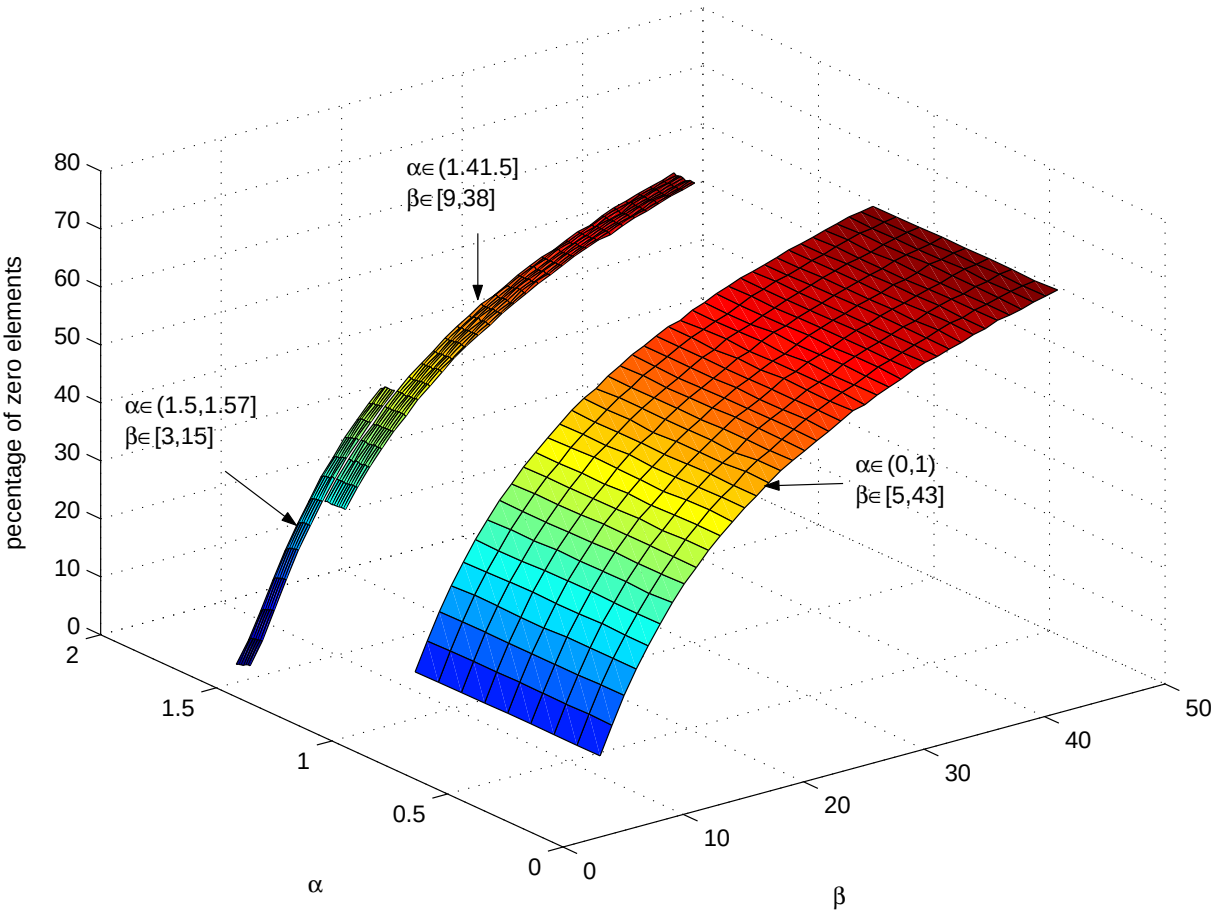


Figure 3: Distribution of zeros in the keystream as a function of (α, β).

cases, too much information is leaked to the attacker, so the cipher becomes totally useless. For the best case, there is still some perceptible visual information existing in the cipher-image, which means that the cipher is not sufficiently secure in all cases.

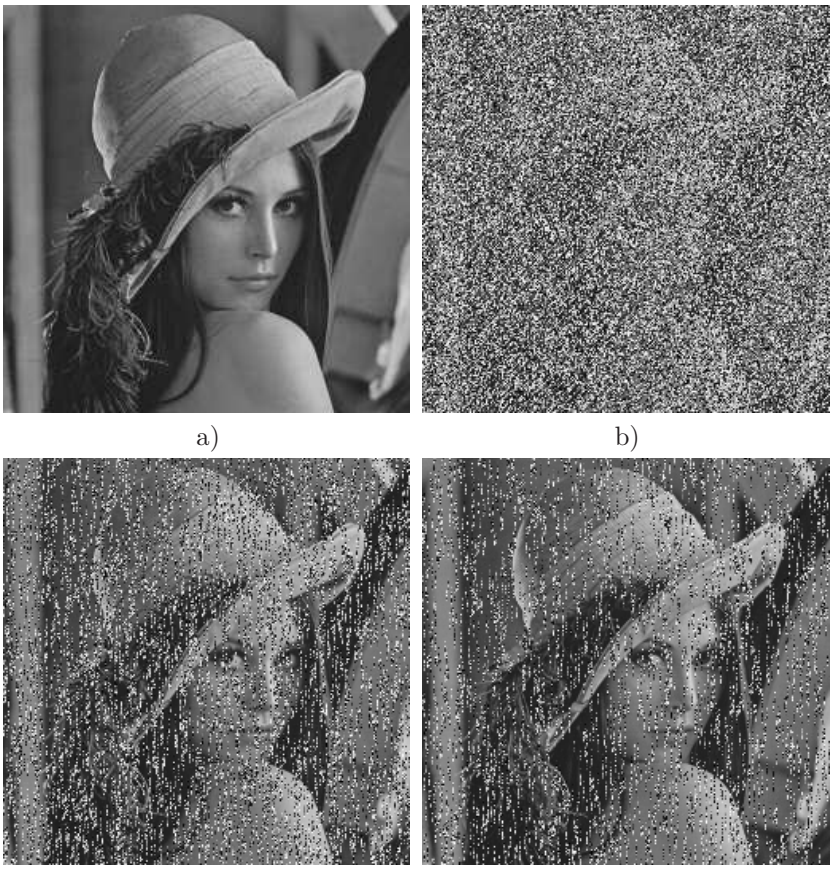


Figure 4: The same image encrypted using three different secret keys: a) original plain-image; b) $K = (\alpha = 1.1,\ \beta = 5,\ x_0 = 0.3)$; c) $K = (\alpha = 1.1,\ \beta = 24,\ x_0 = 0.3)$; and d) $K = (\alpha = 1.1,\ \beta = 43,\ x_0 = 0.3)$.

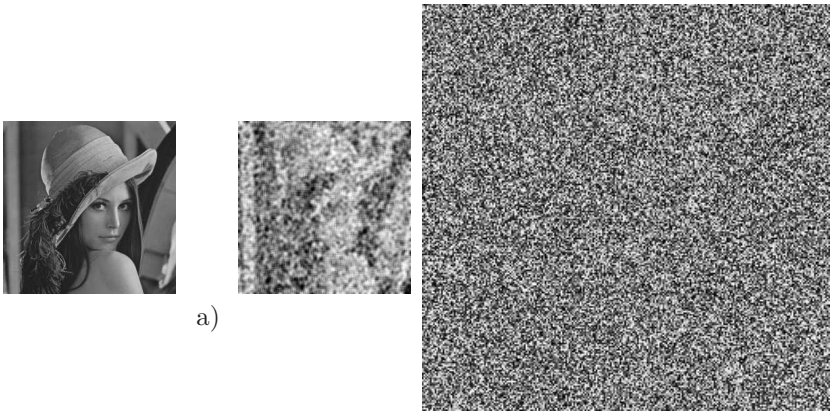
One might be led to think that the cipher-image in Fig. 4.b) looks random enough. However, with a photo manipulation software, one can enhance the image to get more visual information as shown in Fig. 5.a). It should be reminded that 11% of the image pixels remaining unchanged is unacceptable according to general encryption standards.

To minimize the above security problem, (α, β) should be limited within a small area: $\alpha \in (1.5, 1.57], \beta \approx 3$. In this case, it is expected that no major visual information will be leaked, since the percentage of zero bytes is not sufficiently large (see Fig. 5.b) for an example). However, this means a dramatic reduction of the key space by comparison with the original one proposed in [40].

2.2. Plaintext attacks

In the previous subsection we showed that the use of Eq. (1) is not advisable because of its non uniformly distributed orbits. We are to show next that if a different map is used, the security of the communication system will not improve if the same key is used repeatedly for successive encryptions.

According to [42, p. 25], it is possible to differentiate between different levels of attacks on cryptosystems. In a known plaintext attack, the opponent possesses a plain-image, p, and the corresponding encrypted image,



a) b)

Figure 5: Encrypted images: a) retrieved image from Fig. 4.b); best possible encrypted image, obtained when $K = (\alpha = 1.54, \beta = 3, x_0 = 0.3)$.

c. In a chosen plain text, the opponent has obtained temporary access to the encryption machinery, and hence he can choose a plain-image, p, and construct the corresponding encrypted image, c.

The cipher under study behaves as a modified version of the one-time pad [42, p. 50]. The one-time pad uses a randomly generated key of the same length as the message. To encrypt a plain-image p, its pixel bytes are combined with the random key k using the exclusive-OR operation bitwise. Mathematically,

$$c_i = p_i \oplus k_i, \tag{2}$$

where c represents the encrypted image. This method of encryption is perfectly secure because the encrypted message, formed by XORing the message and the random key, is itself totally random. It is crucial to the security of the one-time pad that the key be as long as the message and never reused, thus preventing two different messages encrypted with the same portion of the key being intercepted or generated by an attacker.

In the present cipher, Eq. (1) is used to generate a keystream to encrypt the plain-image according to the rule of Eq. (2). Therefore, if the attacker possesses the plain-image p and its corresponding cipher-image c, he will be able to obtain k. If the same key $K \in \mathcal{K}$, i.e. the same parameter values, is used to encrypt any subsequent message in the future, it will generate an identical chaotic orbit, which is already known. As a consequence, when c and k are known in Eq. (2), p is readily obtained by the attacker.

Obviously, when using this cryptosystem, regardless of the choice of the chaotic map, the key can never be reused. This is a problem peculiar to all stream ciphers, and not just this one: obtaining the keystream produces the same effect as obtaining the key, and thus any subsequent message encrypted with the same key will be broken. Therefore, the claim of unbrealability by chosen/known plaintext attacks in Sec. 5 of [40] is totally unfounded.

As an example, let us consider a black plain-image, i.e. all its bytes are 0. Its encryption using Eq. (2) will yield an encrypted image corresponding to the exact values of the keystream. Any subsequent image encrypted by the same key (which generates the same keystream) of equal or smaller size, will be easily decrypted now that the keystream is known.

Let us note that stream ciphers can never reuse the key because they are all vulnerable to chosen/known plaintext attacks. Hence the suggested rule 11 in [41], which has not been followed:

It should be checked whether the designed cryptosystem can be broken by the relatively simple known-plaintext and chosen-plaintext attacks, and even chosen-ciphertext attacks.

2.3. The key space

It is a common error to relate the security of an encryption algorithm to the size of the key space. Quoting from [43]:

"A necessary, but usually not sufficient, condition for an encryption scheme to be secure is that the key space be large enough to preclude exhaustive search."

Whereas a short key means that the best encryption algorithm can be broken by exhaustive search (also known as brute force attacks) in a reasonable amount of time, the reverse is not true. Hence, claims such as "compared with some general encryption algorithms such as DES, the encryption algorithm is more secure" in the abstract of [40] are totally unfounded. Suggested rule 15 in [41] must always be followed, but does not guarantee security:

To provide a sufficient security against brute-force attacks, the key space size should be $\kappa > 2^{100}$.

In addition, as can be deduced from the observation of Figs. 2 and 3, the key space is greatly reduced because it should be limited to areas of the phase space for which the number of zeros generated in the keystream is statistically well balanced. Unfortunately, the chaotic map in Eq. (1) does not allow for that range and thus cannot be used for cryptographic purposes in the way presented in [40]. The key space $\mathcal{K}$ given by the authors violates the following suggested rules from [41]:

Suggested Rule 5 *The key space* $\mathcal{K}$*, from which valid keys are to be chosen, should be precisely specified and avoid non-chaotic regions.*

Suggested Rule 10 *The ciphertext should be statistically undistinguishable from the output of a truly random function, and should be statistically the same for all keys.*

If some postprocessing operations are added to manipulate the chaotic orbit of the map[1], it is possible to find a way to generate the keystream with a uniform distribution. However, such a study is out of the scope of this cryptanalysis paper.

2.4. A Computational Error in [40]

Finally, it deserves mentioning that the experimental data shown in Fig. 2 of [40] are wrong. This figure was claimed to be generated when $x_0 = 0.3, \alpha = 1.57, \beta = 3.5$. However, our experiments in Matlab and VC++ gave a completely different orbit as shown in Fig. 6, under double-precision floating-arithmetic. We also tested the single-precision implementation, and noticed that the first 20 points of the orbit are almost identical with those shown in Fig. 6. As a result, we believe that the authors of [40] committed an error in their implementation of the NCA map, which should lead to a totally wrong implementation of the image encryption scheme.

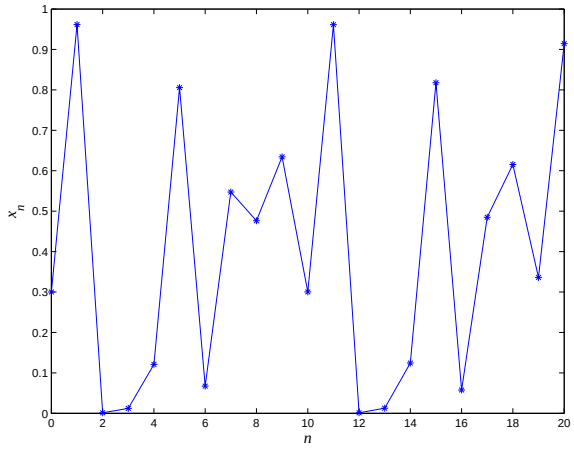


Figure 6: The real chaotic orbit of the NCA map when $\alpha = 1.57$, $\beta = 3.5$ and $x_0 = 0.3$ (compare it with Fig. 2 in [40]).

[1]For example, some bits of each chaotic state are combined in a specific way to produce one or more bytes of the keystream.

3. Conclusions

We have shown that the image encryption scheme proposed in [40] is insecure because it uses a chaotic map with bad statistical properties. The keystream generated by this map, upon which the cipher is built, has a non-uniform distribution, making it inappropriate for cryptographic uses. Following the rules suggested in [41] during the design of the algorithm would have prevented these flaws.

Acknowledgments

This work was supported by *CDTI, Ministerio de Industria, Turismo y Comercio of Spain* in collaboration with Telefónica I+D, Project SEGUR@ with reference CENIT-2007 2004, *CDTI, Ministerio de Industria, Turismo y Comercio of Spain* in collaboration with SAC, project HESPERIA (CENIT 2006-2009), *Ministerio de Ciencia e Innovación of Spain*, project CUCO (MTM2008-02194). Shujun Li was supported by a fellowship from the Zukunftskolleg of the Universität Konstanz, Germany, which is part of the "Exzellenzinitiative" Program of the DFG (German Research Foundation).

References

[1] M. Hasler. Synchronization of chaotic systems and transmission of information. *Int. J. Bifurcation Chaos*, 8:647–659, 1998.

[2] G. Alvarez, F. Montoya, M. Romera, and G. Pastor. Chaotic cryptosystems. In Larry D. Sanson, editor, *Proc. 33rd Annual 1999 International Carnahan Conference on Security Technology*,, pages 332–338. IEEE, 1999.

[3] Christopher P. Silva and Albert M. Young. Introduction to chaos-based communications and signal processing. In *Proc. IEEE Aerospace Conference*,, pages 279–299, 2000.

[4] T. Yang. A survey of chaotic secure communication systems. *Int. J. Comput. Cogn.*, 2(2):81–130, 2004.

[5] Roland Schmitz. Use of chaotic dynamical systems in cryptography. *J. Frankl. Inst.*, 338(4):429–441, 2001.

[6] L. Kocarev. Chaos-based cryptography: A brief overview. *IEEE Circuits Syst. Mag.*, 1(3):6–21, 2001.

[7] S. Li. *Analyese and New Designs of Digital Chaotic Ciphers*. PhD dissertation, School of Electronics and Information Engieering, Xi'an Jiaotong University, Xi'an, China, 2003. available online at `http://www.hooklee.com/pub.html`.

[8] T. Habutsu, Y. Nishio, I. Sasase, and S. Mori. A secret key cryptosystem by iterating a chaotic map. In *Advances in Cryptology – EuroCrypt'91*, volume 547 of *Lecture Notes in Computer Science*, pages 127–140. Spinger-Verlag, Berlin, 1991.

[9] M. S. Baptista. Cryptography with chaos. *Phys. Lett. A*, 240(1-2):50–54, 1998.

[10] E. Álvarez, A. Fernández, P. García, J. Jiménez, and A. Marcano. New approach to chaotic encryption. *Phys. Lett. A*, 263(4-6):373–375, 1999.

[11] Stergios Papadimitriou, Tassos Bountis, Seferina Mavaroudi, and Anastassios Bezerianos. A probabilistic symmetric encryption scheme for very fast secure communications based on chaotic systems of difference equations. *Int. J. Bifurcation Chaos*, 11(12):3107–3115, 2001.

[12] W.-K. Wong, L.-P. Lee, and K.-W. Wong. A modified chaotic cryptographic method. *Comp. Phys. Comm.*, 138(3):234–236, 2001.

[13] L. Kocarev and G. Jakimoski. Logistic map as a block encryption algorithm. *Phys. Lett. A*, 289(4-5):199–206, 2001.

[14] G. Jakimoski and L. Kocarev. Chaos and cryptography: Block encryption ciphers based on chaotic maps. *IEEE Trans. Circuits Syst. I*, 48(2):163–169, 2001.

[15] P. García and J. Jiménez. Communication through chaotic map systems. *Phys. Lett. A*, 298(1):35–40, 2002.

[16] K.W. Wong. A fast chaotic cryptographic scheme with dynamic look-up table. *Phys. Lett. A*, 298(4):238–242, 2002.

[17] K.W. Wong. A combined chaotic cryptographic and hashing scheme. *Phys. Lett. A*, 307(5-6):292–298, 2003.

[18] Xun Yi, Chik How Tan, and Chee Kheong Siew. A new block cipher based on chaotic tent maps. *IEEE Trans. Circuits Syst. I*, 49(12):1826–1829, 2002.

[19] N. K. Pareek, V. Patidar, and K. K. Sud. Discrete chaotic cryptography using external key. *Phys. Lett. A*, 309(1-2):75–82, 2003.

[20] L. Kocarev and Z. Tasev. Public-key encryption based on chebyshev maps. In *Proceedings of the IEEE Symposium on Circuits and Systems (ISCAS'03)*, volume 3, pages 28–31, 2003.

[21] S. Li, G. Chen, and X. Zheng. Chaos-based encryption for digital images and videos. In Borko Furht and Darko Kirovski, editors, *Multimedia Security Handbook*, chapter 4, pages 133–167. CRC Press, LLC, December 2004. preprint is available online at `http://www.hooklee.com/pub.html`.

[22] E. Biham. Cryptoanalysis of the chaotic-map cryptosystem suggested at EuroCrypt'91. In *Advances in Cryptology – EuroCrypt'91*, volume 547 of *Lecture Notes in Computer Science*, pages 532–534. Spinger-Verlag, Berlin, 1991.

[23] G. Alvarez, F. Montoya, M. Romera, and G. Pastor. Cryptanalysis of a chaotic encryption system. *Phys. Lett. A*, 276(1):191–196, 2000.

[24] Shujun Li, Xuanqin Mou, Boliya L. Yang, Zhen Ji, and Jihong Zhang. Problems with a probabilistic encryption scheme based on chaotic systems. *Int. J. Bifurcation Chaos*, 11(10):3063–3077, 2003.

[25] G. Alvarez, F. Montoya, M. Romera, and G. Pastor. Cryptanalysis of a chaotic secure communication system. *Phys. Lett. A*, 306(4):200–205, 2003.
[26] G. Alvarez, F. Montoya, M. Romera, and G. Pastor. Keystream cryptanalysis of a chaotic cryptographic method. *Comp. Phys. Comm.*, 156(2):205–207, 2003.
[27] G. Alvarez, F. Montoya, M. Romera, and G. Pastor. Cryptanalysis of an ergodic chaotic cipher. *Phys. Lett. A*, 311(2-3):172–179, 2003.
[28] G. Alvarez, F. Montoya, M. Romera, and G. Pastor. Cryptanalysis of a discrete chaotic cryptosystem using external key. *Phys. Lett. A*, 319(3-4):334–339, 2003.
[29] G. Alvarez, F. Montoya, M. Romera, and G. Pastor. Cryptanalysis of dynamic look-up table based chaotic cryptosystems. *Phys. Lett. A*, 326(3-4):211–218, 2004.
[30] S. Li, G. Chen, and X. Mou. On the security of the Yi-Tan-Siew chaotic cipher. *IEEE Trans. Circuits Syst. II*, 51(12):665–669, 2004.
[31] Pina Bergamo, Paolo D'Arco, Alfredo De Santis, and Ljupco Kocarev. Security of public-key cryptosystems based on Chebyshev polynomials. *IEEE Trans. Circuits Syst. I*, 52(7):1382–1393, 2005.
[32] Shujun Li and Xuan Zheng. Cryptanalysis of a chaotic image encryption method. In *Proceedings of 2002 IEEE International Symposium on Circuits and Systems (ISCAS'2002)*, pages 708–711, 2002.
[33] Shujun Li and Xuan Zheng. On the security of an image encryption method. In *Proceedings of 2002 IEEE International Conference on Image Processing (ICIP'2002)*, volume 2, pages 925–928, 2002.
[34] Chengqing Li, Shujun Li, Dan Zhang, and Guanrong Chen. Cryptanalysis of a chaotic neural network based multimedia encryption scheme. In *Advances in Multimedia Information Processing - PCM 2004 Proceedings, Part III*, volume 3333 of *Lecture Notes in Computer Science*, pages 418–425. Springer-Verlag, 2004.
[35] Chengqing Li, Shujun Li, Guanrong Chen, Gang Chen, and Lei Hu. Cryptanalysis of a new signal security system for multimedia data transmission. *EURASIP J. Appl. Signal Process.*, 2005(8):1277–1288, 2005.
[36] Chengqing Li, Xinxiao Li, Shujun Li, and Guanrong Chen. Cryptanalysis of a multistage encryption system. In *Proceedings of 2005 IEEE International Symposium on Circuits and Systems (ISCAS'2005)*, pages 880–883, 2005.
[37] Chengqing Li, Shujun Li, Dan Zhang, and Guanrong Chen. Chosen-plaintext cryptanalysis of a clipped-neural-network-based chaotic cipher. In *Advances in Neural Networks C ISNN 2005 Proceedings, Part II*, volume 3497 of *Lecture Notes in Computer Science*, pages 630–636. Springer-Verlag GmbH, 2005.
[38] Chengqing Li, Shujun Li, Dan Zhang, and Guanrong Chen. Cryptanalysis of a data security protection scheme for VoIP. *IEE Proc.-Vis. Image Signal Process.*, 153(1):1–10, 2006.
[39] Chengqing Li, Shujun Li, Der-Chyuan Lou, and Dan Zhang. On the security of the Yen-Guo's domino signal encryption algorithm (DSEA). *J. Syst. Softw.*, 79(2):253–258, 2006.
[40] Haojiang Gao, Yisheng Zhang, Shuyun Liang, and Dequn Li. A new chaotic algorithm for image encryption. *Chaos Solitons Fractals*, 29:393–399, 2005.
[41] G. Alvarez and S. Li. Some basic cryptographic requirements for chaos-based cryptosystems. *Int. J. Bifurcation Chaos*, 16(8):2129–2151, 2006.
[42] D. R. Stinson. *Cryptography: Theory and Practice*. CRC Press, 1995.
[43] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, 1997.