PHYSICAL REVIEW E, VOLUME 65, 045201(R)

Coupled map networks as communication schemes

P. García
Departamento de Física Aplicada, Facultad de Ingeniería, Universidad Central de Venezuela, Caracas, Venezuela

A. Parravano and M. G. Cosenza
Centro de Astrofísica Teórica, Facultad de Ciencias, Universidad de Los Andes, Apartado Postal 26, La Hechicera, Mérida 5251, Venezuela

J. Jiménez and A. Marcano
Laboratorio de Fenónemos no Lineales, Escuela de Física, Facultad de Ciencias Universidad Central de Venezuela, Caracas, Venezuela

(Received 28 August 2001; published 28 March 2002)

Networks of chaotic coupled maps are considered as string and language generators. It is shown that such networks can be used as encrypting systems where the cipher text contains information about the evolution of the network and also about the way to select the plain text symbols from the string associated with the network evolution. The secret key provides the network parameters, such as the coupling strengths.

DOI: 10.1103/PhysRevE.65.045201 PACS number(s): 05.45.Vx, 02.50.−r, 05.45.Gg, 05.45.Ra

Most languages produce aperiodic messages with finite entropy [1]. Since this property is emblematic of chaotic systems, they are potential candidates to model simple languages and to design communication schemes [2–7]. Most of these models and schemes have considered the use of only one chaotic dynamic, either for masking the message to be sent or for transmitting a controlled signal. However, these procedures may result in poor security when used as a communication system [8–10]. On the other hand, a large number of connected physiological units are involved in real languages. Thus, it seems interesting to explore the performance of a network of interacting chaotic elements as a model to generate simple languages and as communication schemes.

In this article we study networks of coupled chaotic maps as generators of strings of symbols, and investigate their potential use as an encrypting system. A coupled map network (CMN) can be defined as

$$x_{t+1}^{i}=f(x_t^i)+\sum_{j=1}^{N}\epsilon_{ij}x_t^j, \tag{1}$$

where x_t^i gives the states of the element i ($i=1,\ldots,N$) at discrete time t; $f(x_t^i)$ is a real function describing the local dynamics; ϵ_{ij} are the coupling strengths among elements in the system; and N is the size of the network. Coupled map lattices have provided fruitful models for the study of a variety of spatiotemporal processes in spatially distributed systems [11].

Equation (1) can be written in vector form as

$$\mathbf{x}_{t+1}=\mathbf{f}(\mathbf{x}_t)+\mathbf{E}\mathbf{x}_t. \tag{2}$$

The state vector $\mathbf{x}_t$ possesses N components $\mathbf{x}_t=(x_t^1,x_t^2,\ldots,x_t^N)$, corresponding to the states of the elements in the network. The $N\times N$ elements of matrix $\mathbf{E}$ are ϵ_{ij}, which we assume in general to be different among themselves, i.e., the coupling is heterogeneous.

The system Eq. (2) can be used as a string generator and thus it can produce a sequence of symbols. For simplicity, we shall consider a CMN consisting of $N=7$ maps. To each CMN state $\mathbf{x}_t=(x_t^1,x_t^2,\ldots,x_t^7)$ we can assign a binary state $(b_t^1\,b_t^2\,b_t^3\,b_t^4\,b_t^5\,b_t^6\,b_t^7)$ by the following rule: $b_t^i=0$ if $x_t^i<x^*$, and $b_t^i=1$ if $x_t^i>x^*$, where x^* is some threshold value. With a prefixed correspondence rule, each of the 128 possible seven-digit binary states $(b_t^1\,b_t^2\,b_t^3\,b_t^4\,b_t^5\,b_t^6\,b_t^7)$ can be associated with one ASCII symbol z_k among the set $Z_{128}=\{z_1,z_2,\ldots,z_{128}\}$. We take $x^*=0$. In this case, each seven-digit binary state corresponds to one of the $2^7=128$ "Cartesian quadrants" in the seven-dimensional state space of the CMN, enough to assign an ASCII symbol z_k ($k=1,2,\ldots,128$) to each "quadrant."

Let us assume that, starting from any initial condition $\mathbf{x}_0$, the state vector of the CMN visits all the "quadrants" during its evolution, so that all ASCII symbols in Z_{128} are generated by the CMN dynamics. If we assign to the state $\mathbf{x}_t$ the ASCII symbol corresponding to the "quadrant" where $\mathbf{x}_t$ lies at time t, the string $\alpha=(z_{k_1},z_{k_2},\ldots,z_{k_t},\ldots,z_{k_T})$ of ASCII symbols will be generated up to time T. We denote by $|\alpha|=T$ the length of the string, i.e., the number of iterations performed on the CMN system up to time $t=T$.

On the other hand, for a given set of ordered ASCII symbols $\rho=(p_1,p_2,\ldots,p_n)$, a sufficiently long string α can be expressed as a succession of substrings $\beta_l\cdot p_l$,

$$\alpha=(\beta_1\cdot p_1,\beta_2\cdot p_2,\ldots,\beta_{l-1}\cdot p_{l-1},\underbrace{\beta_l\cdot p_l}_{l\text{ segment}},\ldots,\beta_n\cdot p_n), \tag{3}$$

where $\beta_1 \cdot p_1$ is the substring beginning at z_{k_1} and ending at the first occurrence of symbol p_1, the substring $\beta_2 \cdot p_2$ begins after p_1 and ends at the first occurrence of symbol p_2, and so on. For example, the string

$$\alpha=(d,4,\$,R,m,e,>,i,\&,H,+,t,5,v,?,u,K,g,a,i,a,6,l) \tag{4}$$

is segmented by the word "Rival," i.e., $\rho=(R,i,v,a,l)$ as

$$\alpha = (\underbrace{d,4,\$}_{\beta_1}, \overbrace{R}^{p_1}, \underbrace{m,e,>}_{\beta_2}, \overbrace{i}^{p_2}, \underbrace{\&,H,+,t,5}_{\beta_3}, \overbrace{v}^{p_3}, \underbrace{?,u,K,g}_{\beta_4}, \overbrace{a}^{p_4}, \underbrace{i,a,6}_{\beta_4}, \overbrace{l}^{p_5}). \tag{5}$$

The set of marker symbols $(p_1,p_2,\ldots,p_l,\ldots,p_n)$ univocally determines how string α is segmented by the rule in Eq. (3).

Any string α resulting from the evolution of the CMN can always be expressed as a concatenation of segments $\beta_l \cdot p_l$, provided that there are no symbols forbidden by the dynamics of the CMN, i.e., all the "quadrants" are visited by the state vector $\mathbf{x}_t$. Let $\mathbf{y}_{l-1}$ be the state of the CMN when the symbol p_{l-1} occurs at the end of substring $\beta_{l-1} \cdot p_{l-1}$. The next substring $\beta_l \cdot p_l$ depends only on $\mathbf{y}_{l-1}$ and on the symbol p_l. Then, the substring $\beta_l \cdot p_l$ can be expressed as

$$\beta_l \cdot p_l = g(\mathbf{y}_{l-1},p_l), \tag{6}$$

where the function g is just the procedure described above to generate strings from the CMN dynamics starting from $\mathbf{x}_0 = \mathbf{y}_{l-1}$ and ending at the first occurrence of symbol p_l. Thus, the function g is the recipe by which a sequence of ASCII symbols is associated with the sequence of state vectors arising from the evolution of the CMN. The autonomous evolution of the CMN system yields the string

$$\alpha = g(\mathbf{x}_0,p_1)\cdot g(\mathbf{y}_1,p_2)\cdot \ldots \cdot g(\mathbf{y}_{l-1},p_l)\cdot \ldots . \tag{7}$$

The segmentation of string α by a finite string $\rho = (p_1,p_2,\ldots,p_n)$ can be represented by an n-dimensional vector $\mathbf{c}(\rho)$ whose components are the natural numbers giving the lengths $|\beta_l \cdot p_l|$. That is,

$$\mathbf{c}(\rho) = (|g(\mathbf{x}_0,p_1)|,|g(\mathbf{y}_1,p_2)|,\ldots,|g(\mathbf{y}_{n-1},p_n)|). \tag{8}$$

Since the CMN can be iterated indefinitely, Eq. (8) just expresses the segmentation $\mathbf{c}(\rho)$ for the first $T = \Sigma_{i=1}^{n} |g(\mathbf{y}_{i-1},p_i)|$ symbols of the string α. In the example given in Eq. (5), one gets $\mathbf{c}(\rho)=(4,4,6,5,4)$.

The segmentation $\mathbf{c}(\rho)$ in Eq. (8) provides the position of the symbols $(p_1,p_2,\ldots,p_n)$ in string α, and therefore it can be used as the encryption of the plain text $\rho = (p_1,p_2,\ldots,p_n)$. That is, after a number of $t = \Sigma_{i=1}^{l} |g(\mathbf{y}_{i-1},p_i)|$ iterations from $\mathbf{x}_0$ the plain text symbol p_l is generated by the CMN evolution. If the local dynamics $f(x)$ is public, the secret key may consist of the couplings ϵ_{ij} and the initial condition $\mathbf{x}_0$. Once the couplings are specified, the autonomous evolution of the CMN will generate a string α that depends only on $\mathbf{x}_0$. In other words, under autonomous evolution, if the matrix $\mathbf{E}$ and $\mathbf{x}_0$ are used as a secret key, the string α will always be the same for a fixed key. Therefore, a number $n=|\rho|$ symbols of string α can be known if the plain text ρ and its corresponding cipher text $\mathbf{c}(\rho)$ are known. Unknown elements between the n known symbols $(p_1,p_2,\ldots,p_n)$ can be inferred by using new messages encrypted with the same key, even when the new plain texts are unavailable. In the example given in Eq. (5), the word "Rival" is encrypted as $\mathbf{c}(R,i,v,a,l)=(4,4,6,5,4)$; therefore, after 19 iterations and after 23 iterations the CMN generates the symbols "a" and "l", respectively. If another word has the encryption $\mathbf{c}(\rho)=(4,4,4,4,3,4)$, it can be guessed that $\rho=(R,i,t,u,a,l)$, and two additional symbols of the string α can be inferred.

Note that this decoding method is possible because, for the autonomous evolution of the CMN, the string α is unique for a given key. In order to avoid this limitation, a nonautonomous CMN evolution can be used. A possibility is to make the string α dependent on the plain text to be encrypted. We call this method text dependent encryption (TDE). An example of TDE is a CMN dynamic that is perturbed each time a symbol is encrypted. This perturbation could be, for example, a change of sign in the states of the maps x_t^i each time that a symbol p_l is encrypted. In this case, when the plain text $\rho=(p_1,p_2,\ldots,p_n)$ is encrypted, the resulting CMN evolution string α can be expressed as

$$\alpha(\rho) = g(\mathbf{x}_0,p_1)\cdot g(-\mathbf{y}_1,p_2)\cdot \ldots \cdot g(-\mathbf{y}_{n-1},p_n), \tag{9}$$

and the corresponding encryption is

$$\mathbf{c}(\rho) = (|g(\mathbf{x}_0,p_1)|,|g(-\mathbf{y}_1,p_2)|,\ldots,|g(-\mathbf{y}_{n-1},p_n)|). \tag{10}$$

Note that the lth segment $\beta_l \cdot p_l$ of the string α is univocally determined by the previous $(l-1)$ symbols in the plain text ρ. Conversely, the encryption $\mathbf{c}(\rho)$ in Eq. (10) allows one to reproduce the CMN dynamics generated during the encryption process, since $\mathbf{c}(\rho)$ indicates at which iteration steps the dynamics must be perturbed. Therefore, both the string $\alpha(\rho)$ and the plain text ρ can be reconstructed if the appropriate key is used (i.e., the appropriate CMN parameters and initial condition). We call this encryption method TDE$(*-1)$, to indicate that the CMN vector state is multiplied by (-1) each time that a symbol is encrypted. The encryption with autonomous CMN evolution can be denoted by TDE$(*+1)$. Other operations can be used in the TDE method.

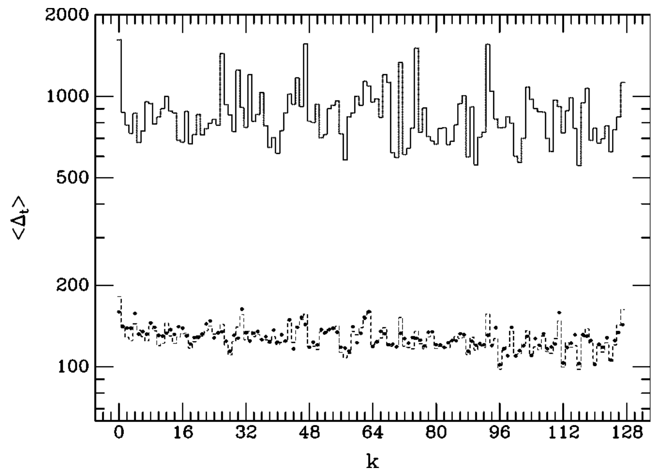

FIG. 1. Mean distance between successive occurrences of the same symbol z_k in an autonomous string α of length $|\alpha|=50\,000$ as a function of k $(k=1,2,\ldots,128)$, for fixed $b=0.47$. When ρ is a string consisting of a repetition of the same symbol z_k (i.e., $\rho=z_k,z_k,\ldots\equiv\overline{z_k})$, the dots give the average $\langle\Delta_t\rangle\equiv\langle|\beta_l\cdot z_k|\rangle$ length of segments in $\mathbf{c}(\overline{z_k})$ [see Eq. (8)]. The dashed curve shows the standard deviation of the segment lengths in $\mathbf{c}(\overline{z_k})$. The upper dotted curve displays the maximum values of the segment lengths $\Delta_t(\overline{z_k})$ in the 50 000 iterations; the minimum segment lengths lie between 1 and 4.

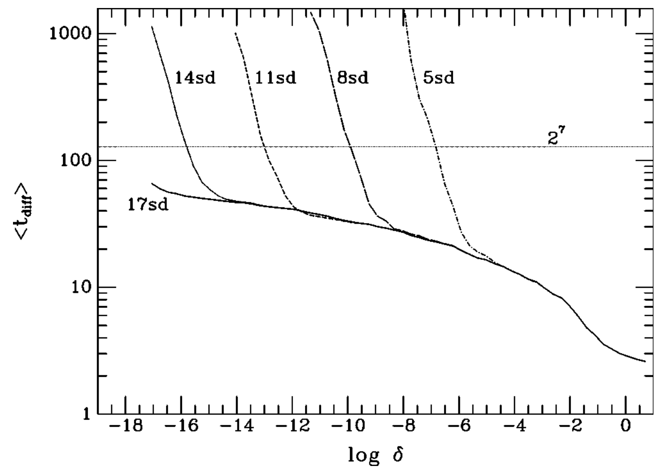

FIG. 2. Mean number of iterations $\langle t_{diff}\rangle$ as a function of the logarithm in base 10 of the size of the perturbation δ, for $b=0.47$. The various curves correspond to different truncations of the CMN states after each iteration t. The value $\langle t_{diff}\rangle$ is obtained by averaging the 7×7 results $t_{diff}(\delta_{ij}=\delta)$ for $i,j=1,2,\ldots,7$. The labels indicate the number of significant digits of the components of $\mathbf{u}_t$ (i.e., 17sd indicates 17 significant digits).

In principle, any aperiodic function $f(x)$ can be used as a local map in the CMN system, Eq. (1). As an example, we consider the unbounded local chaotic dynamics given by the logarithmic map [12] $f(x)=b+\ln|x|$. This map is chaotic, with no periodic windows, in the parameter interval $b\in(-1,1)$. The unbounded character of the local functions places no restrictions on the range of parameter values of the CMN system that can be explored. For local parameter values about $b\approx0.5$, and couplings randomly selected in the interval $|\epsilon_{ij}|<0.1$, $\forall i,j$, all the ASCII symbols in the set Z_{128} are generated by the seven-dimensional CMN with about the same probability of 1/128, as can be seen in Fig. 1. This shows that all the 2^7 "Cartesian quadrants" are visited by the state vector of the CMN in about 2^7 iterations. Note also that the standard deviation of the substring lengths are of the same order of magnitude as their average, which is typical of an aperiodic string.

Another useful property of chaotic CMNs as encrypting schemes is their sensitivity to initial conditions and/or couplings. The sensitivity to the couplings can be measured by comparing two strings α and α' generated by two CMNs identical to each other, except by one element in their coupling matrices, $\epsilon'_{ij}=\epsilon_{ij}+\delta_{ij}$. Figure 2 shows the mean number of iterations $\langle t_{diff}\rangle$ at which the strings α and α' start to differ, as a function of the size of the perturbation δ_{ij}. The various curves correspond to different truncations of the CMN states after each iteration t. The truncation used consists of expressing the real value of each component of the state vector $\mathbf{x}_t$ with a given number of significant digits. The truncated state $\mathbf{u}_t$ is used to calculate the state $\mathbf{x}_{t+1}$ at iteration $t+1$. That is, $\mathbf{x}_{t+1}=\mathbf{f}(\mathbf{u}_t)+\mathbf{E}\mathbf{u}_t$. This truncation is relevant since it can be used to make the numerical process equivalent in computers with different precision.

Since the typical number of iterations to find a given symbol is about 2^N, we measure the encrypting sensibility δ_{cri} of the CMN as the value of δ for which $\langle t_{diff}\rangle=2^N$. Note that δ_{cri} is a very small value ($\sim10^{-12}$ for a 10-significant digit truncation).

For a fixed value of the parameter b and $N=7$, the maximum encrypting key consists of 7×7 coupling strengths and seven initial conditions x_0^i. As shown in Fig. 2, a change of $\delta=10^{-10}$ in one of the coupling strengths is more than enough to modify the string α after $t\approx40$. Therefore, there are more than $10^{\delta^{-1}\times N\times(N+1)}\sim10^{560}$ possible keys. Obviously, among all of these possibilities there are groups of keys that produce strings α that are identical to each other up to $t_{diff}\gg40$, but the probability of finding two such keys is very small ($\sim2^{-Nt_{diff}}$). In general, such a large number of possible keys is unnecessary, and in practice the key can be reduced by using a set of random number seeds that are used to generate the 7×7 coupling strengths and the seven initial conditions x_0^i. Alternatively, the system size N can be reduced in order to decrease the number of possible keys and to increase the encrypting speed.

As an example, for $b=0.5$, $x_0^i=1.0+0.1i$, and $\epsilon_{ij}=0.01(i-j/2)$ $(i,j=1,\ldots,7)$, the encryption of the text "Rival ritual" would be

(a) using the autonomous CMN evolution [Eq. (8)],

128 44 18 530 33 505 7 206 97 95 8 170;

(b) using the encryption method TDE($*-1$) [Eq. (10)],

128 387 64 34 36 3 96 297 146 26 78 3;

(c) using the encryption method TDE($*-1$) [Eq. (10)], but adding the small quantity 10^{-10} to the coupling weight $\epsilon_{3\,5}$,

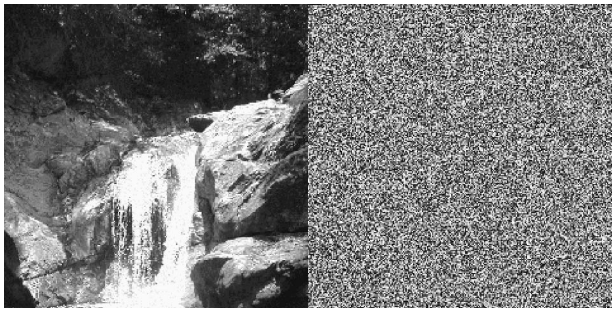

FIG. 3. The BMP (bitmap) image on the left has been encrypted assuming $b=0.5$, $x_0^i=1.0+0.1i$, and $\epsilon_{ij}=0.01(i-j/2)$ $(i,j=1,\ldots,8)$. On the right, the corresponding decoded image is shown when a slightly wrong key is used: we have added 10^{-10} to ϵ_{35}.

$$425\ 176\ 20\ 156\ 8\ 85\ 234\ 43\ 32\ 87\ 224\ 80.$$

If we try to recover the plain text using the encryption method TDE$(*-1)$ in (b), but using the coupling $\epsilon_{3\,5}$ altered by the amount $\delta_{3\,5}=10^{-10}$, the resulting decoded text is

$$0<3w7h\$|lR''.$$

In the above example, the FORTRAN internal function ICHAR has been used to assign a binary seven-digit number to each of the ASCII symbols in the set Z_{128}.

Notice that, using $N=8$, the CMN dynamics can be employed to generate strings with elements in a palette of 256 gray tones and therefore to encrypt images byte by byte, as shown in Fig. 3.

In conclusion, we have shown how a CMN can be used as a string generator and as an encrypting system. The cipher text contains the information on how the CMN must be evolved and how to select the plain text symbols from the string associated with the CMN's evolution. The secret key consists of the coupling matrix **E** and the initial conditions. The number of parameters involved in the secret key and the high sentivity of the generated strings to small perturbations of any of those parameters make the CMN encrypting scheme difficult to break. This confers an advantage on this scheme, in terms of security, in comparison to communication procedures based solely on one chaotic dynamic. The use of several coupled dynamics instead of just one allows the transmission of entire sequences of the plain text at a time. The notation introduced allows one to place the proposed encrypting method in a wide context. The implementation of variations of the method is straightforward. The examples presented here show the encrypting performance for a network of seven coupled logarithmic maps; however, the method can be applied with networks of any size. Finally, we note that the CMN parameters determine both the probability of occurrence of symbols and the transition probability $p(z_j|z_i)$ of observing symbol z_i followed by symbol z_j in the string. Therefore, it is possible in principle to select the CMN parameters in order to enhance or to inhibit some symbols and transitions. Since this is equivalent to selecting grammatical rules, the CMN as string generators might be of interest in the development of language models.

M.G.C. and A.P. acknowledge support from Consejo de Desarrollo Científico, Humanístico y Tecnológico of Universidad de Los Andes, Mérida, Venezuela. The authors thank Professor R. Pino for useful comments.

[1] C. E. Shannon and W. Weaver, *The Mathematical Theory of Communication* (University of Illinois Press, Chicago, 1964).
[2] M. Baptista, E. Rosa, and C. Grebogi, Phys. Rev. E **61**, 3590 (2000).
[3] M. Baptista, E. Macau, C. Grebogi, Y. Lai, and E. Rosa, Phys. Rev. E **62**, 4835 (2000).
[4] I. Mariño, E. Rosa, and C. Grebogi, Phys. Rev. Lett. **85**, 2629 (2000).
[5] E. Alvarez, A. Fernández, P. García, J. Jiménez, and A. Marcano, Phys. Lett. A **263**, 373 (1999).
[6] S. Hayes, C. Grebogi, E. Ott, and A. Mark, Phys. Rev. Lett. **73**, 1781 (1994).
[7] G. Van Wiggeren and R. Roy, Science **279**, 1198 (1998).
[8] G. Alvarez, F. Montoya, M. Romera, and G. Pastor. Phys. Lett. A **276**, 191 (2000).
[9] G. Perez and H. Cerdeira, Phys. Rev. Lett. **74**, 1970 (1995).
[10] T. Yang, L. Yang, and C. Yang, Phys. Lett. A **245**, 495 (1998).
[11] Chaos **2**, 279 (1992), focus issue on coupled map lattices, edited by K. Kaneko.
[12] T. Kawabe and Y. Kondo, Prog. Theor. Phys. **85**, 759 (1991).