ELSEVIER

25 March 2002

Physics Letters A 295 (2002) 185–191

PHYSICS LETTERS A

www.elsevier.com/locate/pla

Channel coding in communications using chaos

Inés P. Mariño *, Luis López, Miguel A.F. Sanjuán

Nonlinear Dynamics and Chaos Group, Universidad Rey Juan Carlos, 28933 Móstoles, Madrid, Spain

Received 14 November 2001; received in revised form 1 February 2002; accepted 1 February 2002

Communicated by C.R. Doering

Abstract

We introduce a novel chaotic channel code with error-correcting capabilities. This channel code takes advantage of the natural redundancy contained in the perturbations applied to a chaotic system, in order to encode a desired message in the symbolic dynamics of the chaotic waveform. © 2002 Elsevier Science B.V. All rights reserved.

PACS: 05.45.+b

Keywords: Channel coding; Error-correcting codes; Chaotic systems; Controlling chaos

1. Introduction

Recent developments in communicating with chaos have provided a great variety of potential practical applications, which include transmitter-receiver synchronization [1–4], signal masking and recovery [5,6], noise filtering [7], encryption [5], reconstruction of information signals [8,9] and encoding/decoding algorithms that allow to embed an arbitrary digital message into the symbolic dynamics of a chaotic system [8–14]. The latter contributions show that it is possible to guide the evolution of a chaotic signal by applying small perturbations on the system variables. This feature allows to generate controlled chaotic waveforms whose symbolic representation corresponds to a desired message. Thus, these techniques are specially appealing because they take advantage of the most outstanding property of chaotic systems, their extreme sensitivity to the initial conditions, which had been previously seen as an obstacle for practical applications. Moreover, this control approach has already successfully provided new solutions for some classical problems in digital communications, such as the design of a robust transmission system in the presence of impulsive noise [8,9].

In this Letter, we elaborate on the control technique reported in Ref. [8] and make a first effort to investigate its potential application to the important task of channel coding [15]. Channel coding consists of deliberately introducing redundancy in the transmitted signal in a way that enables the receiver to detect and, sometimes, correct the bit errors caused by channel noise and/or distortion. Although relatively little attention has been devoted to this topic from the point of view of chaotic dynamics, a remarkable contribution is the paper by Baptista et al. [16], where a communication scheme fully based on chaos theory is proposed, including the implementation of a channel dynami-

* Corresponding author.
E-mail address: iperez@escet.urjc.es (I.P. Mariño).

0375-9601/02/$ – see front matter
PII: S0375-9601(02)00146-9

cal encoder. We would also like to mention the work by Chen and Wornell [17], where interesting results on the design of channel codes using discrete-time chaotic systems are shown, as well as their connection to conventional coding methods. In this Letter, a different approach is introduced that consists of exploiting the natural redundancy of a continuous-time chaotic signal that bears a desired message within its symbolic dynamics. As a result, a novel error-correcting code is proposed, whose performance is illustrated through computer simulations.

In Section 2 we review the control technique of [8] and discuss some details, basically dealing with the robustness of the method, that had not been studied before and are relevant to the error-correcting code proposed in Section 3. Finally, Section 4 contains some final remarks and conclusions.

2. Controlling the symbolic dynamics of a chaotic system

As shown in [8,10,11], small perturbations applied on the system trajectory of a chaotic attractor can be used to make the output waveform carry a desired symbol sequence representing a message.

The chaotic Lorenz system,

$$\begin{aligned}\dot{x} &= -\sigma(x-y),\\ \dot{y} &= Rx - y - xz,\\ \dot{z} &= bz + xy,\end{aligned} \tag{1}$$

provides a good framework for investigating the idea of controlling dynamics with small perturbations to the system trajectory. Recall that, for the standard parameter values, $\sigma = 10$, $R = 28$, and $b = 8/3$, the state coordinate $(x(t), y(t), z(t))$ moves on a chaotic attractor in a three-dimensional state space forming two lobes. The standard parameters will be used all throughout the Letter. In order to control the system, let us also consider the two Poincaré surfaces of section given by the half-planes $y = \pm\sqrt{b(R-1)}$ and $|x| \geqslant \sqrt{b(R-1)}$, each one defined on a different lobe (see Fig. 1). When the system crosses the surface with $y = -\sqrt{b(R-1)}$ in a previously fixed direction it is said to generate a symbol "A" and when the system crosses the surface with $y = +\sqrt{b(R-1)}$ it is said to generate a symbol "B". Thus, a direct relationship exists between the system time evolution and the symbol string resulting from the successive crossings, which is referred to as the *symbolic dynamics* of the system.

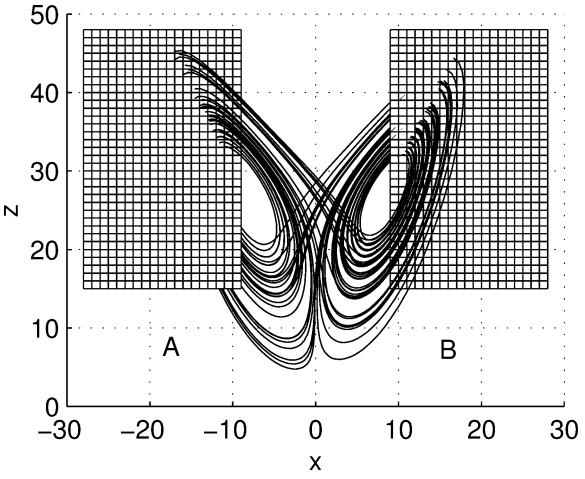

Fig. 1. Projection onto the x–z plane of the chaotic Lorenz attractor crossed by the Poincaré surfaces defined by $y = +\sqrt{b(R-1)}$ and $|x| \geqslant \sqrt{b(R-1)}$ and by $y = -\sqrt{b(R-1)}$ and $|x| \geqslant \sqrt{b(R-1)}$. These surfaces are represented, respectively, by the symbols "A" and "B".

Since the system is deterministic, there is a one-to-one correspondence between the point where the three-dimensional state coordinate crosses one of these surfaces of section and the future n-symbol sequence, $s_1 \ldots s_n$, generated after the crossing. The first symbol, s_1, indicates the present crossing and s_n represents the surface that is being crossed $n-1$ oscillations later. When the system runs free, the long-term temporal evolution of the state coordinate yields a random-like symbol string. This is easily observed in the scalar continuous-time signal $x(t)$ shown in Fig. 2, where the symbols "A" and "B" appear as a random-like sequence of positive and negative peaks.

In Ref. [8], a control technique was first introduced that allows to determine the symbol string generated by the chaotic system after crossing a Poincaré surface of section. Such a control is exercised by applying small perturbations to one single system variable, z, instead of the complete state coordinate, thus leading to a simple implementation. Let us remark that, in this context, *control* means the ability to guide the system so that it generates a desired string of symbols carrying a message (instead of a random-like sequence, as the uncontrolled system does) while its overall evolution remains chaotic.

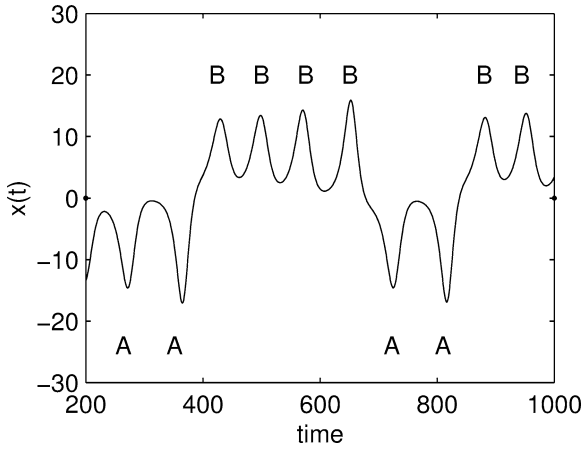

Fig. 2. Temporal evolution of the variable x of the Lorenz system.

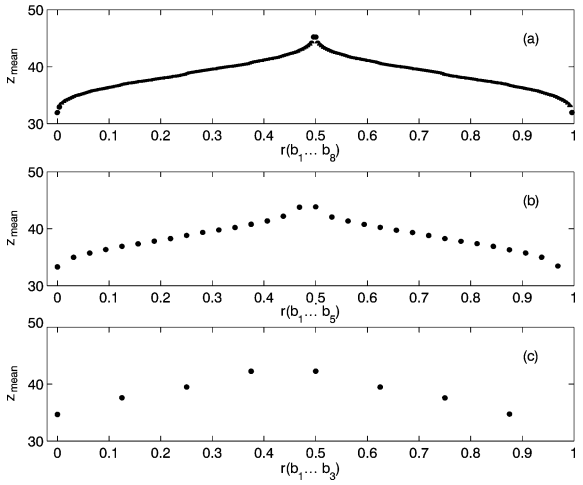

Fig. 3. Values of z_{mean} associated with each different n-symbol sequence, represented by $r(s_1 \dots s_n)$: (a) for $n = 8$, (b) for $n = 5$, and (c) for $n = 3$.

In order to know how to apply small perturbations on $z(t)$, a learning process is needed. Let us assume that we want to control the n-symbol string $s_1 \dots s_n$ generated after a crossing. The learning process is characterized by the value of n and works as follows. We let the system run freely for a long enough period of time and observe both the values that variable $z(t)$ takes at the crossings with the surfaces of section and the subsequent n-symbol strings generated by the system. In that way, we can associate a range of values of z, hereinafter referred to as a *bin*, to each one of the 2^n different possible n-symbol sequences. Once the bins are identified, an average of all z in the same bin, referred to as z_{mean}, can be calculated (notice that all z in the same bin have been observed to generate the same n-symbol sequence). The value of z_{mean} associated to each bin completely determines the future n-symbol sequence, represented by the real magnitude $r(s_1 \dots s_n) = \sum_{i=1}^{n} f(s_i) 2^{-i} < 1$, where $f(A) = 0$ and $f(B) = 1$. When this association is done for all bins, the learning process is complete: this is all the information the controller requires, as shown later.

As already mentioned, the learning process is characterized by the value of n. Thus, it is interesting to study how this magnitude affects the relationship between z_{mean} and $r(s_1 \dots s_n)$. Although, for a fixed value of n, the 2^n different bins do not have the same length we can define an average size of the bin, that we call $S(n)$. This average size decays approximately as $S(n) = 36 \cdot 2^{-n}$ (see [8]). However, it is more important to watch the evolution of the relationship between z_{mean} and $r(s_1 \dots s_n)$ as n varies. This is plotted for $n = 8$, $n = 5$ and $n = 3$ in Fig. 3(a), (b) and (c), respectively. We can observe how values of z_{mean} corresponding to neighbouring sequences for the three cases are farther from each other as n decreases. This is due to the fact that the size of each bin associated with each different sequence increases as n decreases. It is also apparent from this figure that different n-symbol sequences beginning with the same symbol are associated with strictly different values of z_{mean}.

The procedure to control the symbolic dynamics is examined next. Let us suppose that we want the chaotic waveform $x(t)$ to represent some binary message, as for example,

$$m = 00111010111010011011100 \dots .$$

We can associate the bit "0" with a change of symbol in two consecutive crossings with the Poincaré surfaces of section (i.e., AB or BA) and the bit "1" with the repetition of the same symbol (i.e., AA or BB). In this way, the same message can be represented either by the following symbol sequence,

$$ABAAAABBAAABBABBBAAAABB \dots ,$$

that begins with the symbol "A", or by the sequence

$$BABBBBAABBBAABAAABBBBAA \dots ,$$

that begins with the symbol "B". This is useful so as to avoid problems related to the symmetry of the $z_{\text{mean}}(r)$ function (see Fig. 3: every value of $z_{\text{mean}}(r)$ is associated with two complementary binary sequences) and is usually referred to as a *differential* encoding in the context of digital communications [15], where the problem of decoding an information signal consisting of pulses with an unknown phase rotation is usually encountered. Hence, the information is not extracted from the absolute phase of the pulses but from the phase difference between consecutive pulses. In our chaotic scheme, we have to face a similar problem, as explained in the next section.

The aim of the encoder is to introduce small perturbations in the variable $z(t)$ at each crossing with the surfaces of section in order to generate the desired n-symbol sequence. This desired sequence, $s_1 \dots s_n$, consists of $n-1$ symbols which are predetermined plus one new information symbol, i.e., s_1 is given by the current crossing and $s_2 \dots s_{n-1}$ must be the same symbols that would be generated by the system if we did not apply any perturbation at all. Therefore, the perturbation we apply in the kth crossing sets the value of the $(k+n)$th symbol generated by the system, the perturbation in the $(k+1)$th crossing sets the $(k+n+1)$th symbol and so on. Notice that the larger the value of n, the smaller the perturbations and the more the system time evolution resembles the one of a free system.

From a practical point of view, each perturbation is chosen to move the z coordinate to the central point in the bin corresponding to the desired n-bit sequence, i.e., the z_{mean} value. An alternative procedure, maybe more rigorous, would be to move the z coordinate to the point that yields the highest likelihood of generating the sequence, but our numerical simulations have proved that moving to the central point is a rather good approximation. Actually, the fact of moving the z coordinate to any of the spots inside the bin associated to the desired sequence would be a potentially good choice which should not be neglected, especially for *short term* encoding, i.e., for low values of n. The reason is that, according to the expression $S(n) = 36 \cdot 2^{-n}$, we have larger bins for smaller n, implying that there is a wider range of values of z that may yield the desired evolution of the chaotic system. A corollary of this argument is that if the practical, possibly electronic, device we use for introducing the perturbations is subject to some type of noise, then short term (small n) encoding should be implemented in order to guarantee a robust performance.

3. A novel error-correcting channel code

3.1. Communication scheme

Once a desired message is encoded into the chaotic waveform $x(t)$ the most straightforward design of a communication system consists of transmitting this chaotic signal $x(t)$ through a communication channel. At the receiver, the message could be recovered just by observing the sequence of positive and negative peaks of the variable $x(t)$, possibly corrupted by thermal noise and other sources of distortion. This approach has already been described in the literature [8,9], where signal reconstruction methods are proposed to account for the existence of impulsive noise in the channel by exploiting the properties of the encoding method and the chaotic signal. Here, we explore a different approach where the signal $x(t)$ is not transmitted itself. Instead, the transmission part of the communication system is implemented by conventional engineering methods, while the controlled Lorenz system is used to construct a novel *channel code*, i.e., a redundant representation of the message to be transmitted [15], that enables the receiver to partially detect and correct the transmission errors caused by channel noise and other sources of distortion.

Due to the control procedure that has been described to encode the desired message into the chaotic waveform $x(t)$, it is apparent that, knowing the initial conditions for the Lorenz system, $x(0)$, $y(0)$, $z(0)$, with a sufficient degree of accuracy, all the relevant information of the signal $x(t)$ (and, therefore, the message) is contained in the value of the perturbations applied to the variable $z(t)$ at each crossing with a Poincaré surface. Hence, we propose the communication scheme depicted in Fig. 4 where only the values of the successive spots $z_{\text{mean}}(k)$, $(k = 0, 1, \dots)$, that indicate where the variable $z(t)$ must be placed by the control algorithm, are transmitted. Notice that index $k = 0, 1, \dots$ represents the crossing or symbol number.

The communication system takes the following successive steps.

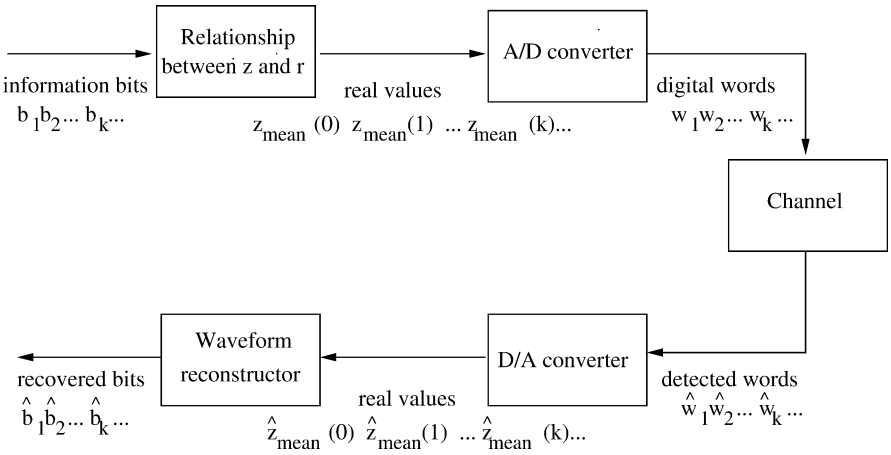

Fig. 4. Communication scheme for the proposed error-correcting channel code.

- The information bits (i.e., the message), $b_1 b_2 \ldots b_k \ldots$, are converted into a sequence of real values $z_{\text{mean}}(0) z_{\text{mean}}(1) \ldots z_{\text{mean}}(k) \ldots$ using the learned relationship between variable z and the symbolic dynamics of the system.
- A conventional analog-to-digital converter (A/D) transforms this sequence of real values into digital *words*, $w_1 w_2 \ldots w_k \ldots$. Each of these words is a signal in digital format that may be easily transmitted through the communication channel (see [15] for details on analog-to-digital conversion and digital modulation). This is a classical problem that can be solved in several different ways using well-tested engineering methods.
- A conventional digital receiver detects the digital words. Let us refer to the *detected* words as $\hat{w}_1 \hat{w}_2 \ldots \hat{w}_k \ldots$. The reason to use a different notation is that digital detection is subject to errors due to channel noise and distortion, hence, the detected word, $\hat{w}_k$, may be different from the transmitted one, w_k.
- A conventional digital-to-analog (D/A) converter transforms the detected words into a real sequence $\hat{z}_{\text{mean}}(0) \hat{z}_{\text{mean}}(1) \ldots \hat{z}_{\text{mean}}(k) \ldots$, where $\hat{z}_{\text{mean}}(k) = z_{\text{mean}}(k)$ if, and only if, $\hat{w}_k = w_k$.
- The real sequence $\hat{z}_{\text{mean}}(0) \hat{z}_{\text{mean}}(1) \ldots \hat{z}_{\text{mean}}(k) \ldots$ is used to reconstruct the temporal evolution of the variable $x(t)$ from a perturbed Lorenz system using the control algorithm described in Section 2. If no errors occurred during the transmission of the digital words, the recovered message, $\hat{b}_1 \hat{b}_2 \ldots \hat{b}_k \ldots$ will coincide with the original one, $b_1 b_2 \ldots b_k \ldots$. Notice that the recovered message is observed in the peaks of the variable $x(t)$ of the reconstructed perturbed system.

Overall, the proposed communication scheme can be seen as *splitting* the control algorithm into two parts: at the transmitter, we take the message and compute the perturbations (actually, the $z_{\text{mean}}(k)$ values) to be applied on the Lorenz system. This information regarding the perturbations is *passed* to the receiver conventionally, meaning that we use standard digital communication techniques. At the receiver, we apply the perturbations on the Lorenz system and observe the time evolution of the variable $x(t)$ in order to recover the message.

At this point, it is easy to explain the reason to employ a differential encoding of the information as described in Section 2. Let us consider for a while that we had taken the more straightforward approach of encoding the message directly on variable $x(t)$ as $A = 1$ and $B = 0$. Then, we use the communication scheme described above and observe the message on the time evolution of the variable $x(t)$ at the receiver. Let us assume that, at some time, $\hat{w}_k \neq w_k$ and, consequently, $\hat{z}_{\text{mean}}(k) \neq z_{\text{mean}}(k)$. If the difference $\hat{z}_{\text{mean}}(k) - z_{\text{mean}}(k)$ is large enough, there will be a symbol error in the $(k+1)$th crossing, i.e., we observe 1 when the original bit was 0 or vice versa. This

means that the system trajectory has moved to the *wrong* lobe. Then, if there are no further errors and we keep applying the correct perturbations, $\hat{z}_{\text{mean}}(k+p) = z_{\text{mean}}(k+p)$, $p = 1, 2, 3\ldots$, we will observe a symbolic sequence of 1s and 0s that is the complement (that changes 0s by 1s and vice versa) of the original message. The reason is the symmetry of the function $z_{\text{mean}}(r)$ (as shown in Fig. 3) which implies that every value z_{mean} is associated with two complementary n-symbol strings.

This drawback is easily avoided by using differential encoding because the information is not represented by the symbols themselves, but by the transition from one symbol to the next. Hence, as explained in Section 2, both a symbol string and its complement represent the same message.

3.2. Protection from errors

Since the perfect recovery of the message at the receiver using the scheme described above depends on whether there are errors or not in the conventional digital transmission step, the obvious question is: why is this scheme better than simply transmitting the information bits $b_1 b_2 \ldots b_k \ldots$ conventionally? The answer is that the proposed form of transmission turns out to provide protection against transmission errors because the $z_{\text{mean}}(k)$ values are highly redundant. Indeed, if the perturbations are small enough, the deterministic behavior of the system allows to predict, from the value of the variable $z(t)$ at any crossing with the Poincaré surfaces of section, which symbols will be generated in the $n-1$ subsequent crossings. This is the basis of the encoding method. As explained in the previous section, the small perturbation applied in the kth crossing with a Poincaré surface modifies the symbol $s(k+n-1)$ produced by the system, but symbols $s(k)\ldots s(k+n-2)$ are the same as if the perturbation had not been applied.

What is the effect of this property on the receiver? Recall that $z_{\text{mean}}(k)$ is the central value of the bin associated with the n-symbol string generated after the kth crossing with a surface of section. Therefore, even if there are some mismatch, i.e., if $\hat{z}_{\text{mean}}(k) \neq z_{\text{mean}}(k)$ due to transmission errors, $\hat{z}_{\text{mean}}(k)$ may still be within the bin associated with the same n-symbol string as $z_{\text{mean}}(k)$ and we will still recover the same information without error. But more importantly, even if $\hat{z}_{\text{mean}}(k)$ does not belong to the same bin as $z_{\text{mean}}(k)$, it is likely to belong to a neighbouring bin which, by construction, is associated to a symbol sequence that only differs in the last symbols. Hence, errors can still be avoided.

We clarify this point with an example. Consider a system with 5-symbol encoding and assume that $z_{\text{mean}}(k)$ belongs to bin number 3, associated with the string 00010, where the first symbol, 0, represents the kth crossing, the second symbol, 0, represents the $(k+1)$th crossing, and so on, up to the $(k+4)$th crossing (symbol 0). By construction, bins number 2 and 4 are associated to strings 00001 and 00011. Hence, if $\hat{z}_{\text{mean}}(k)$ belongs to any of these three bins the symbol associated with the $(k+1)$th crossing will be the same.

3.3. A simple example

In order to clarify how the receiver can work we show the following example for 3-symbol control. At the transmitter, there is a local Lorenz oscillator with standard parameters that is controlled using the procedure in Section 2 to differentially encode a desired binary message. After the kth crossing (with $k = 0, 1, 2, \ldots$), the resulting value of $z_{\text{mean}}(k)$ is converted into a binary word, w_k, which is digitally transmitted. Just to illustrate the method (a more sophisticated algorithm should be used in practice) we add a single parity bit to w_k (this yields w_k, p_k as the new word to be transmitted). This parity bit allows to easily detect transmission errors of one bit. When no error is detected, the received word is converted back into a real form to yield the corresponding $z_{\text{mean}}(k)$ value, which is used to control the local Lorenz system. When an error is detected, no perturbation is applied at the kth crossing of the oscillator and we let the redundancy of the system to account for this absence. Notice that the *previous* perturbation already fixed the next two symbols, so we are not actually losing information unless two perturbations in a row are absent. Clearly, this is the most straightforward way of implementing the proposed channel coding method, but it serves to the purpose of illustrating its error-correcting capability.

The performance of a system using the described channel code is shown in Fig. 5. We plot the coded Bit Error Rate (BER), which is the BER attained

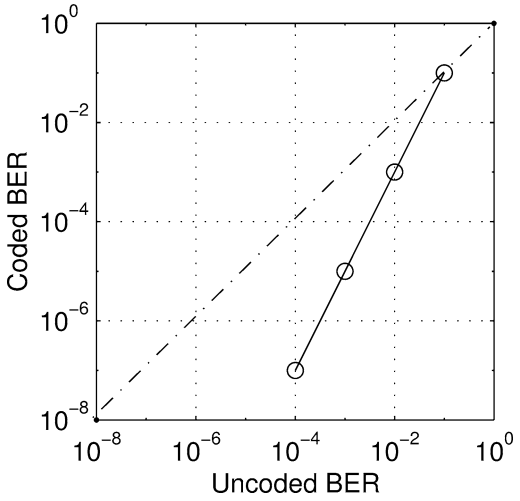

Fig. 5. Coded BER as a function of the uncoded BER when 3-symbol control is used.

by the system when the proposed channel code with $n = 3$ is applied versus the natural BER of the digital channel, i.e., the BER of the binary transmission system when no coding is used neither to detect nor to correct errors. This will be called uncoded BER. In the plot, the diagonal line represents the performance of the uncoded system. After decoding, points over the diagonal indicate a performance loss, meaning that the BER has worsened, and values below the diagonal indicate a performance improvement, i.e., a reduction in BER after channel decoding. We observe that a gain of up to three orders of magnitude is achieved when the uncoded BER is 10^{-4}.

4. Conclusions

The most significant characteristic of chaotic systems is their great sensitivity to small perturbations. It is well known that this characteristic allows to guide the trajectory of this type of dynamical systems by the proper application of small perturbations directly on the system variables at some strategically chosen points on the trajectory. This procedure can be employed to differentially encode an arbitrary binary message within a continuous-time chaotic waveform. This chaotic waveform can be considered as an information-bearing signal that naturally presents a high degree of redundancy. In fact, all the redundancy is contained in the applied perturbations. Thus, we have exploited this fact to introduce a novel chaotic channel code with error-correcting capabilities. The performance of this chaotic channel code has been illustrated through computer simulations for the case of the Lorenz system.

To conclude we would like to remark that, although the goal of this Letter is not secure communication, the fact of transmitting through the communication channel the value of the applied perturbations, instead of the chaotic waveform itself, can provide some advantages since the receiver needs to know the chaotic system and the appropriate parameters to reconstruct the information-bearing chaotic waveform.

Acknowledgements

This work was supported by the Ministry of Science and Technology, Spain, under project BFM2000-0967.

References

[1] L.M. Pecora, T.L. Carroll, Phys. Rev. Lett. 64 (1990) 821.
[2] K.M. Cuomo, A.V. Oppenheim, Phys. Rev. Lett. 71 (1993) 65.
[3] L. Kocarev, U. Parlitz, Phys. Rev. Lett. 74 (1995) 5028.
[4] M. Hasler, Int. J. Bifurcation Chaos 8 (1998) 647.
[5] K.M. Short, Int. J. Bifurcation Chaos 7 (1997) 1579.
[6] M.S. Baptista, Phys. Lett. A 240 (1998) 50.
[7] E. Rosa Jr., S. Hayes, C. Grebogi, Phys. Rev. Lett. 78 (1997) 1247.
[8] I.P. Mariño, E. Rosa Jr., C. Grebogi, Phys. Rev. Lett. 85 (2000) 2629.
[9] I.P. Mariño, C. Grebogi, E. Rosa Jr., Int. J. Bifurcation Chaos 11 (2001) 2631.
[10] S. Hayes, C. Grebogi, E. Ott, Phys. Rev. Lett. 70 (1993) 3030.
[11] S. Hayes, C. Grebogi, E. Ott, A. Mark, Phys. Rev. Lett. 73 (1994) 1781.
[12] D. Gligoriski, D. Dimovski, L. Kocarev, V. Urumov, L.O. Chua, Int. J. Bifurcation Chaos 6 (1996) 2119.
[13] E. Bollt, Y.-C. Lai, C. Grebogi, Phys. Rev. Lett. 79 (1997) 378.
[14] E. Bollt, Y.-C. Lai, Phys. Rev. E 58 (1998) 1724.
[15] J.G. Proakis, Digital Communications, McGraw–Hill, Singapore, 1995.
[16] M.S. Baptista, E.E. Macau, C. Grebogi, Y.-C. Lai, E. Rosa Jr., Phys. Rev. E 62 (2000) 4835.
[17] B. Chen, G.W. Wornell, IEEE Trans. Commun. 46 (1998) 881.