Evaluation of channel coding and decoding algorithms using discrete chaotic maps

Francisco J. Escribano[a)]
Nonlinear Dynamics and Chaos Group, Departamento de Matemáticas y Física Aplicadas y Ciencias de la Naturaleza, Universidad Rey Juan Carlos, C/ Tulipán, s/n, 28933 Móstoles, Madrid, Spain

Luis López
Laboratorio de Algoritmia Distribuida y Redes (LADyR), Departamento de Informática, Estadística y Telemática, Universidad Rey Juan Carlos, C/ Tulipán, s/n, 28933 Móstoles, Madrid, Spain

Miguel A. F. Sanjuán
Nonlinear Dynamics and Chaos Group, Departamento de Matemáticas y Física Aplicadas y Ciencias de la Naturaleza, Universidad Rey Juan Carlos, C/ Tulipán, s/n, 28933 Móstoles, Madrid, Spain

(Received 28 April 2005; accepted 30 September 2005; published online 12 January 2006)

In this paper we address the design of channel encoding algorithms using one-dimensional nonlinear chaotic maps starting from the desired invariant probability density function (pdf) of the data sent to the channel. We show that, with some simple changes, it is straightforward to make use of a known encoding framework based upon the Bernoulli shift map and adapt it readily to carry the information bit sequence produced by a binary source in a practical way. On the decoder side, we introduce four already known decoding algorithms and compare the resulting performance of the corresponding transmitters. The performance in terms of the bit error rate shows that the most important design clue is related not only to the pdf of the data produced by the chosen discrete map: the own dynamics of the maps is also of the highest importance and has to be taken into account when designing the whole transmitting and receiving system. We also show that a good performance in such systems needs the extensive use of all the evidence stored in the whole chaotic sequence. © *2006 American Institute of Physics*. [DOI: 10.1063/1.2126807]

The use of nonlinear dynamics and chaos theory to develop new improvements in communications technology has spawned a number of research fields. One of them tries to apply discrete nonlinear chaotic dynamical systems to encode binary and nonbinary signals. Even though some achievements have been described in this field, there are still many aspects to solve and clarify before the described systems can find applicability in real systems. One of the pending topics we address in this paper is the design of practical transmitters with desired statistical properties to develop channel encoding algorithms, based upon a well-known chaotic channel encoding model. Our study provides some hints as to which parameters should be taken into account when facing such design procedures and how to look into the possibility of improving this kind of system in future developments.

I. INTRODUCTION

The possibility of using chaotic signals to carry information was first proposed in 1993[1] and, since then, chaotic communications have been a very important topic in both nonlinear science and engineering. The interest in chaotic communications is related to the good properties of the signals produced by chaotic systems from the point of view of designing secure systems or broadband multiple access systems. In the case of secure systems, one can take advantage of the uncorrelation and unpredictability of the chaotic signals to build encryption algorithms. In practice, however, these expectations of improving secure communications have not been yet fully confirmed. On the other hand, chaotic modulations and channel encoders derived from chaotic systems attracted much attention, due to the possibility of employing them in multiuser systems as efficient means to share the access to the channel, but the interest on chaotic communications dropped somewhat due to the bad performance of the systems proposed so far[2,3] because they did not outperform other usual coded communications schemes and they did not have even better performance than uncoded systems such as binary phase shift keying (BPSK). Nevertheless, in later times we have witnessed the arising of some proposals outperforming classical communications systems,[4–12] and this has opened the way to look further into the possibilities of chaotic systems to act efficiently, for example, as channel encoders and decoders. That is the reason why we address in this paper the evaluation of several decoding algorithms for a given chaos-channel encoding process, in order to shed light on how the overall performance can be improved.

The design of channel coding and decoding algorithms based upon discrete chaotic maps is one of the possibilities studied during the last years, and the relationship between the probability density function (pdf) of the data created and the expression of the map is one of the main topics whose study is required when facing such systems. The task of de-

a)Electronic mail: francisco.escribano@urjc.es

signing a desired map starting from a desired pdf has been successfully addressed[13,14] for the cases we study, though the task of finding the pdf of the data knowing the map remains quite a difficult topic.[15] The knowledge of the pdf, in any case, can help greatly in evaluating the performance of the possible systems, and thus the procedure of designing a map knowing the pdf is of the highest interest. The trouble lies is the design of the practical *encoding algorithm* itself, in such a way as to get a useful transmitter model. As we will show throughout this paper, one of the possibilities is to take advantage of an already known encoding algorithm based upon the Bernoulli shift map and, using the properties of conjugacy between maps, expand it to a whole class of discrete chaotic maps. The design of the map will require some restrictions regarding the form of the pdf and the characteristics of the final map, which will have to be conjugated to the Bernoulli shift map. The importance of this design procedure lies in the fact that such a device has been thoroughly studied for piecewise linear maps (maps, which generate what is called piecewise linear Markov maps),[16,17] while for nonlinear maps, as we do here, there is still a need for a deep study.

According to this aim, we introduce in the second section the encoding process, showing how to build the whole framework. In the third section, we present the four decoding algorithms which serve us to analyze the possibilities of the encoding maps evaluated. The next section is devoted to the presentation of the simulation results and some related analytical results which shed light on the possibilities of the encoding and decoding algorithms. The last section deal with the conclusions and a prospective of future work in this field. In the Appendix, finally, we include an example of the procedure to design a map starting from its desired invariant probability density function, which is used along with the maps introduced in the second section.

II. ENCODING

In order to encode the binary sequence containing the information $\{b_n\}$, where $b_n \in \{0,1\}$ and $n=1\cdots N$, we make use of one method extensively used,[9,12,18,19] which employs an effect of truncation which will be explained in the sequel. The sequence $\{b_n\}$ is a binary sequence, independently and equiprobably distributed, with $p_0=p_1=1/2$. The maps we are interested in are limited to the interval $[0,1]$, and they are the Bernoulli shift map

$$x_{n+1}=f(x_n)=\begin{cases}2x_n & \text{if } x_n<\frac{1}{2}\\ 2x_n-1 & \text{if } x_n\geqslant\frac{1}{2},\end{cases} \tag{1}$$

and what we call the modified logistic map

$$x_{n+1}=f(x_n)=\begin{cases}4x_n(1-x_n) & \text{if } x_n<\frac{1}{2}\\ 1-4x_n(1-x_n) & \text{if } x_n\geqslant\frac{1}{2},\end{cases} \tag{2}$$

which are depicted in Fig. 1.

It is a well-known property of the Bernoulli shift map[20] that, if we define the symbolic state of the system r as

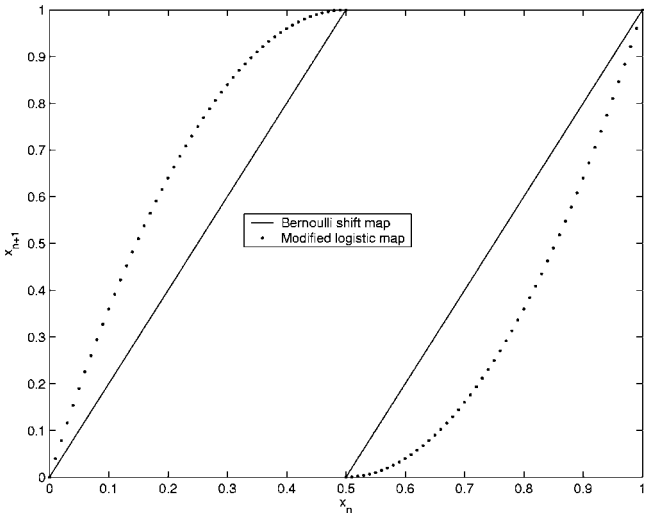

FIG. 1. The figure shows a diagram of the Bernoulli shift and the modified logistic maps.

$$r=\sum_{n=1}^{N} b_n 2^{-n}, \tag{3}$$

and we define the initial condition for the chaotic sequence as $x_0=r$, then the binary sequence is encoded into the chaotic sequence generated by x_0 and the information can be retrieved following the expression:

$$b_n=\left\lfloor x_n+\tfrac{1}{2}\right\rfloor, \tag{4}$$

where $\lfloor x\rfloor$ represents the maximum integer just below x. Similarly, in the case of the modified logistic map, if

$$x_0=\cos^2\left[\frac{\pi}{2}(1-r)\right], \tag{5}$$

then the same property stated in Eq. (4) holds. The function $x_0=g(r)$, whenever it exists for a specific map, makes it possible for the chaotic sequence generated by the corresponding map to carry the information bit sequence following Eq. (4). If the maps are conjugated, as it is the case of the Bernoulli shift map and the modified logistic map, the function $g(r)$ exists and is related to the invariant density $p(x)$ of such map through[14] $g(r)=F^{-1}(r)$, where $F(x)=\int_0^x p(x)dx$ is the distribution function of the invariant density.

For a real system, where the length N of the message, even when sent in packet mode, could reach thousands of bits, the proposed encoding process is not practical, since it implies an almost infinite precision. In this case, the method can be used by encoding blocks of D bits, in the following form:

$$\begin{aligned} r_n' &= \sum_{m=n}^{n+D-1} b_m 2^{-m+n-1},\\ x_n' &= g(r_n'), \end{aligned} \tag{6}$$

where $g(r)$ is the identity in the case of the Bernoulli shift map. This method was proposed originally for the Bernoulli shift map,[9,12,18,19] but we expand it here for the whole class

of conjugated maps. The details and the reasons for this will be shown along with the example drawn in the Appendix. The important point here is to show how to develop a practical transmitting system without the need of infinite precision when using maps other than the Bernoulli shift one. As it may be readily seen, the resulting truncated sequence $\{x_n'\}$ is close to the original one, in a process equivalent to the addition of noise in order to control the chaotic sequence

$$x_n' = x_n + \eta_n, \tag{7}$$

where $\eta_n = O(2^{-D})$ is approximately a random white noise whose power decreases with increasing D. In the sequel, we will refer to the controlled sequence simply as $\{x_n\}$.

The probability density function (or the invariant density) of the data $\{x_n\}$ is uniform for the Bernoulli shift case,[15] $p(x)=1$ when $x \in [0,1]$, whereas for the modified logistic map the pdf is the same as in the case of the logistic map itself, namely[15]

$$p(x) = \frac{1}{\pi\sqrt{x(1-x)}}, \quad x \in [0,1]. \tag{8}$$

We consider here antisymmetric maps with respect to $x = 1/2$ instead of symmetric ones, as the tent and the logistic maps, due to the poor *minimum distance* properties of the latter.[19] The symmetric and the antisymmetric maps maintain the same pdf, though, due to the known relationship met by the related pdfs $p(x)$[15]

$$p(x) = \left| \frac{p[f_1^{-1}(x)]}{f'[f_1^{-1}(x)]} \right| + \left| \frac{p[f_2^{-1}(x)]}{f'[f_2^{-1}(x)]} \right|, \tag{9}$$

where $f_1^{-1}(x)$ and $f_2^{-1}(x)$ are the respective two inverse values obtained through the corresponding map. The absolute value in the expression gives reason of the equal pdfs for both symmetries.

The distribution function of the modified logistic map will thus be $F(x) = 1 - (2/\pi)a\cos(\sqrt{x})$ and it can be verified that $g(r) = F^{-1}(r)$ as stated before. The same holds for the Bernoulli shift map. It is important to notice how, in the case of conjugacy between maps, we can use all these properties to design a map starting from the desired pdf and build a practical transmitter using the truncation shown in the expressions (6).

In the Appendix, following the properties of conjugacy,[14] we design a new map, which we call *flipped logistic pdf* map (FLP map). We will use this map along with the Bernoulli shift and the modified logistic maps in our simulations.

III. DECODING

We have seen how the transmitter side works and now we are ready to look into the channel model, as well as into the decoding process. In the case of the channel model we have chosen, the sequence arriving to the receiver side, $\{y_n\}$, will be

$$y_n = x_n + n_n, \tag{10}$$

where n_n is an additive white Gaussian noise (AWGN) with zero mean and power σ^2. As it is well known, the pdf of this noise is

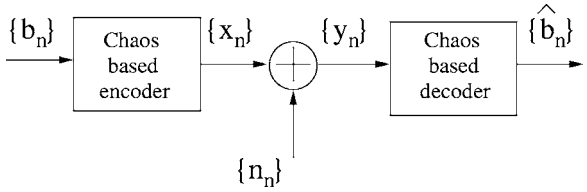

FIG. 2. Block diagram of the described system, consisting on a chaos-based encoder, an AWGN channel, and a chaos-based decoder.

$$p(n) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{n^2}{\sigma^2}}. \tag{11}$$

This is a useful and well-known channel model in telecommunications which makes abstraction of other effects introduced by the usual channels. The complete model of the communications system can be seen in Fig. 2.

In order to take advantage of the good properties of the generated chaotic sequence, a suitable decoding method has to be proposed. First of all, we make use of an already proposed method,[18] with some important modifications that further enhance the performance of the whole system, making it approach the limit performance shown both by theory (in the case of the Bernoulli shift map)[19] and by the following simulations. This will show how the evidence carried by the whole sequence has to be taking into account in the decoding process.

It is clear that the decoding can be made in a symbol-by-symbol basis, by directly deciding over each y_i (hard decoding)[21]

$$\begin{aligned} y_n < \tfrac{1}{2} &\rightarrow \hat{b}_n = 0, \\ y_n \geqslant \tfrac{1}{2} &\rightarrow \hat{b}_n = 1. \end{aligned} \tag{12}$$

Nevertheless, as will be shown next, with this method no gain is achieved with respect to the direct transmission of the source message $\{b_n\}$. Since the pdf of all the maps is known, it is straightforward to calculate the expressions for the probability of error in each case, taking into account that

$$p_e = P\left(y_n < \tfrac{1}{2}, x_n \geqslant \tfrac{1}{2}\right) + P\left(y_n \geqslant \tfrac{1}{2}, x_n < \tfrac{1}{2}\right). \tag{13}$$

By symmetry

$$\begin{aligned} p_e &= 2P\left(y_n \geqslant \frac{1}{2}, x_n < \frac{1}{2}\right) \\ &= 2P\left(n_n \geqslant \frac{1}{2} - x_n, x_n < \frac{1}{2}\right) \\ &= 2\int_0^{1/2} p(x) \int_{1/2-x}^{\infty} p(n)\,dn\,dx \\ &= \int_0^{1/2} p(x)\,\mathrm{erfc}\left[\left(\frac{1}{2} - x\right)\frac{1}{\sqrt{2}\sigma}\right] dx, \end{aligned} \tag{14}$$

where we have used the known relation[21]

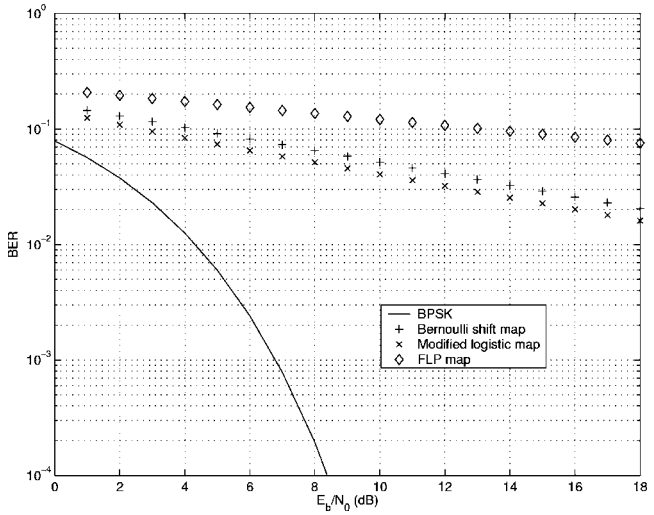

FIG. 3. BER against E_b/N_0 in the case of direct decoding of the y_n sequence for each of the possible encoders based upon the Bernoulli, the modified logistic and the FLP maps. The BER of BPSK (Ref. 21) is shown for comparison.

$$\int_x^\infty p(n)dn = \frac{1}{2}\,\mathrm{erfc}\left(\frac{x}{\sqrt{2}\sigma}\right). \tag{15}$$

The resulting probability of error can be calculated analytically only for the Bernoulli shift map after some algebra, while for the other two maps it has to be calculated by numerical integration. The final expressions are

$$p_e^B = \frac{1}{2}\,\mathrm{erfc}\left(\sqrt{\frac{3E_b}{N_0}}\right) + \frac{1}{\sqrt{12\pi\dfrac{E_b}{N_0}}}\left(1 - e^{-3E_b/N_0}\right) \tag{16}$$

$$p_e^M = \int_0^{1/2} \frac{1}{\pi\sqrt{x(1-x)}}\,\mathrm{erfc}\left[(1-2x)\sqrt{2\frac{E_b}{N_0}}\right]dx \tag{17}$$

$$p_e^F = \int_0^{1/2} \frac{1}{\pi\sqrt{\left(x+\frac{1}{2}\right)\left(\frac{1}{2}-x\right)}} \times \mathrm{erfc}\left[(1-2x)\sqrt{\frac{2}{3-\frac{8}{\pi}}\frac{E_b}{N_0}}\right]dx, \tag{18}$$

where E_b/N_0 is the signal-to-noise ratio in terms of energy per bit against power spectral density of the noise. In our case, $E_b/N_0 = P/(2\sigma^2)$, where P is the power of the sequence and σ^2 is the power of the AWGN process. P is the variance of the sequence x_n, which in the case of the Bernoulli shift map, for D high enough to make the quantization noise negligible, is $\sigma_x^2 = 1/12$; in the case of the modified logistic map, $\sigma_x^2 = 1/8$; and, in the case of the FLP map, $\sigma_x^2 = 3/8 - 1/\pi$. The expression p_e^B corresponds to the Bernoulli shift map, p_e^M to the modified logistic map and p_e^F to the FLP map. The results are depicted in Fig. 3. It is clear in this case that the best is to employ binary signaling BPSK, since, with this scheme, the distance between symbols is maximized. One of the reasons to include the BPSK case throughout this paper, though being an uncoded system, lies in the fact that it can be seen as a limit case of the systems under study, with its pdf collapsed into two delta functions located in $x=0$ and $x=1$. In this limit case, the relationship between the samples is broken since they are fully independent. This hints to a complex tradeoff between the pdf shape and the relationship between consecutive samples that could be worth studying, maybe through a minimum distance analysis as the one made for some linear chaotic maps.[19]

Turning to the performance in terms of the bit error rate (BER), the worst system is the based on the FLP map, since the highest number of samples lies around the threshold point, whereas the modified logistic map slightly improves the performance of the Bernoulli shift map, as there is a reduced density of samples around the threshold point 1/2.

We can also see which is the effect of a finite D in the system, since the last calculations were made assuming $D \to \infty$. In fact, for the Bernoulli shift map, the pdf will be

$$p(x) = \sum_{i=0}^{2^D-1} \frac{1}{2^D}\,\delta\left(x - \frac{i}{2^D}\right), \tag{19}$$

meaning that each of the 2^D samples occur with probability $1/2^D$ (for D high enough, this pdf tends to the uniform pdf within the interval $[0,1]$ under the tight constraint of maintaining the unity area). Taking into account Eq. (13), the real probability of error in the case of simple decoding will be

$$p_e = \frac{1}{2^D}\sum_{i=0}^{2^{D-1}-1} \frac{1}{2}\,\mathrm{erfc}\left[\left(\frac{1}{2}-\frac{i}{2^D}\right)\frac{1}{\sigma\sqrt{2}}\right] + \frac{1}{2^D}\sum_{i=2^{D-1}}^{2^D-1}\left\{1 - \frac{1}{2}\,\mathrm{erfc}\left[\left(\frac{1}{2}-\frac{i}{2^D}\right)\frac{1}{\sigma\sqrt{2}}\right]\right\}, \tag{20}$$

where σ^2 is the power of the noise. To put this expression in terms of E_b/N_0, it should be taken into account that the signal power in this case will be

$$\sigma_x^2 = \frac{1}{12}\frac{2^{2D} + 2^{D+3} - 3}{2^{2D}}.$$

In any case, for $\sigma \to 0$, the only term left will be the one with $i = 2^{D-1}$, which makes $1 - (1/2)\mathrm{erfc}(\cdot) = 1/2$ with independence of the noise power, and the whole p_e tends to an error floor of $1/2^{D+1}$.

This result has to be taken into account when designing the system, since, for some E_b/N_0 beyond the reaching of the limit where $1/2^{D+1}$ becomes dominant, the performance will be substantially different from the performance drawn in the analysis made under the assumption of a continuous and uniformly distributed pdf. In practice, a moderate value between $D=15$ and $D=20$ will be enough, as this pushes the error floor E_b/N_0 beyond the 40 dB. Though drawn for the Bernoulli shift map, similar results can be found for the modified logistic and the FLP maps.

Decoding algorithms

The advantage we can see in the system in a first approach is the redundancy sent in adjoining symbols, since each symbol x_n has $D-1$ bits in common with x_{n-1} and x_{n+1}, $D-2$ with x_{n-2} and x_{n+2}, and so on. The first decoding method, which we call *heuristic decoding*, takes advantage of the property of discrete chaotic systems, where two trajectories starting from points very close could easily diverge, while, on the contrary, two trajectories iterated backwards through the map starting from two different points could finally merge.[18] Therefore, to decode the symbol y_n, we look ahead $M-1$ symbols, and rebuild the possible trajectories that have as ending point y_{n+M-1}. The proposed maps have two possible solutions for the inverse problem, so, as a whole, we will have 2^{M-1} possible trajectories ending in y_{n+M-1}. For example, for $M=2$ and the Bernoulli shift map, we have

$$z_n^0 = \frac{y_{n+1}}{2}, \qquad z_n^1 = \frac{y_{n+1}}{2} + \frac{1}{2}, \tag{21}$$

where, previously, the values of y_{n+M-1} have been normalized within the interval $[0,1]$ for stability reasons, as the maps are defined just over this interval

$$y_{n+M-1} > 1 \rightarrow y_{n+M-1} = 1, \qquad y_{n+M-1} < 0 \rightarrow y_{n+M-1} = 0. \tag{22}$$

In the case of the modified logistic map and $M=2$, the inverse possible values are calculated as

$$z_n^0 = \tfrac{1}{2} - \tfrac{1}{2}\sqrt{1-y_{n+1}}, \qquad z_n^1 = \tfrac{1}{2} + \tfrac{1}{2}\sqrt{y_{n+1}}. \tag{23}$$

For the FLP map and $y_{n+1} < 1/2$

$$z_n^0 = \tfrac{1}{2}\sqrt{\tfrac{1}{2} - \tfrac{1}{2}\sqrt{1-4y_{n+1}^2}}, \qquad z_n^1 = 1 - \tfrac{1}{2}\sqrt{\tfrac{1}{2} + \tfrac{1}{2}\sqrt{1-4y_{n+1}^2}}. \tag{24}$$

For the FLP map and $y_{n+1} \geqslant 1/2$

$$z_n^0 = \tfrac{1}{2}\sqrt{\tfrac{1}{2} + \tfrac{1}{2}\sqrt{1-4(1-y_{n+1})^2}}, \qquad z_n^1 = 1 - \tfrac{1}{2}\sqrt{\tfrac{1}{2} - \tfrac{1}{2}\sqrt{1-4(1-y_{n+1})^2}}. \tag{25}$$

We denote the resulting sequences as $\{z_k^{\mathbf{s}_l}\}$, where $k=n\cdots n+M-2$, $l=1\cdots 2^{M-1}$ and $\mathbf{s}_l$ is the vector defining the lth trajectory, in the form, for example, $(01100\cdots01)$, with 0 and 1 in each case identifying the inverse in the kth position [with the meaning implied in the expressions (21), (23)–(25)].

To decide which is the best candidate $z_n^{\mathbf{s}_l}$ over all possible trajectories $\mathbf{s}_l$, we choose the one whose trajectory is closest to the received one. That is, take $z_n^{\mathbf{s}_{d_n}}$ which makes

$$\sum_{k=n}^{n+M-2} \left(z_k^{\mathbf{s}_{d_n}} - y_k\right)^2 \leqslant \sum_{k=n}^{n+M-2} \left(z_k^{\mathbf{s}_m} - y_k\right)^2, \tag{26}$$

over all possible $m=1\cdots 2^{M-1}$ different trajectories, with $d_n \in \{1\cdots 2^{M-1}\}$. The received bit $\hat{b}_n$ is then decoded following

$$z_n^{\mathbf{s}_{d_n}} < \tfrac{1}{2} \rightarrow \hat{b}_n = 0, \qquad z_n^{\mathbf{s}_{d_n}} \geqslant \tfrac{1}{2} \rightarrow \hat{b}_n = 1. \tag{27}$$

As it may be seen, this algorithm takes advantage of the redundancy present between y_n and the following M symbols, but it does not take into account the redundancy present between y_n and the whole sequence, and this, as we will verify, is of the highest importance. In order to solve this, the algorithm can be applied recursively. Once the sequence of best $z_n^{\mathbf{s}_{d_n}}$ has been calculated for each $n=1\cdots N$, the same algorithm as described can be applied substituting y_n by its corresponding $z_n^{\mathbf{s}_{d_n}}$. It is evident that, in this way, the redundancy present in the whole sequence, from y_n to the end, is propagated backwards and can help to get better results. This second algorithm is what we call the *recursive heuristic* algorithm. It could be deduced that these algorithms will do reasonably well for limited noise, since, in this case, it is more probable that the received sequence $y_i\cdots y_{n+M-1}$ is close to $x_n\cdots x_{n+M-1}$, and the sequence $z_k^{\mathbf{s}_{d_n}}$, with $k=n\cdots n+M-2$ chosen as the closest to the received one, remains closest as well to the sent one. Nevertheless, the simulation results will show that there is an error floor which depends on the decoding complexity (the length of the decoding block M and the number of iterations).

The next decoding method is a *maximum likelihood* (ML) method based upon the Viterbi algorithm,[22] and the last one is a *maximum a posteriori* (MAP) one, based upon the BCJR algorithm.[23] The ML method is an already known one,[24] but with a sliding window framework. The MAP method based upon the BCJR algorithm is straightforwardly adapted from the principles explained in the sequel and it is also applied in a sliding window fashion. For sake of brevity, only the general framework is reviewed here; the details of the algorithms are fully developed in the references cited.

The ML and the MAP algorithms work on a symbolic dynamics basis,[16,17] and so a quantization of the phase space $[0,1]$ is needed. The interval $[0,1]$ is then split into a series of non overlapping intervals I_i with limits i/P and $(i+1)/P$ for $i=0,\cdots,P-1$ and center in $c_i=i/P+1/2P$. P is the number of intervals, taken as an even number (in practice, however, P is also a power of 2 to make the decoding through ML and MAP algorithms feasible), so that the threshold point $1/2$ is the upper point of one interval and the lower one of another. In this way, with the only knowledge that a point x_k lies in interval I_i, we can ascertain whether it has to be decoded as a 1 or as a 0. If we substitute the original sequence by the sequence of intervals where the corresponding symbol lies, we get a symbolic representation of the sequence that can be described as a first order Markov process, with a corresponding transition matrix $\mathbf{T}$. The term t_{ij} in this matrix means the transition probability between interval i

and interval j. In the case of the Bernoulli shift map, each interval maps exactly into two contiguous intervals with equal probability (which is proportional to the length of the original interval that maps to the destination interval). For example, in the case of $P=4$, this transition matrix is

$$\mathbf{T} = \frac{1}{2}\begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}.$$

In the case of the modified logistic and the FLP maps, the transition matrix is not so straightforward to be derived, and it has to be built by considering the quotient between the length of the intersection between the mapped interval $f(I_i)$ and the destination interval I_j to the length of the image $f(I_i)$, i.e.,

$$t_{ij} = \frac{\text{length}[f(I_i) \cap I_j]}{\text{length}[f(I_i)]}. \tag{28}$$

We then consider the sequence d_k^i, $k=0\cdots L-1$, where $d_k^i = c_i$ if x_k lies in I_i. L is the length of the block of symbols considered to decode one symbol in the sliding window scheme. In the same way, we say that the Markov process is in state $s_k=i$ at time $k=1\cdots L$ if x_k lies in I_i. The starting state s_0 is considered to be initially 0, while, as the sliding window proceeds with the decoding, the initial state is taken from the previously calculated values. Under all these assumptions, it is relatively easy to adapt both the Viterbi and the BCJR algorithms for each of the proposed maps.[24]

IV. SIMULATION RESULTS AND FURTHER ANALYSIS

In Figs. 4–6 and Figs. 9–13 we can see the results obtained with the proposed algorithms for different parameters. In all cases, we have chosen $D=20$ bits per symbol. This guarantees that 2^{-D} is small enough to make the difference between x_n and x_n' negligible in practice, while making it possible to use the encoding scheme with an arbitrary block size N.

As we may see in Figs. 4–6, where the results with the heuristic algorithm are shown, the BER obtained improves as M becomes larger. We can also see two different regions in the resulting BER graphs: one waterfall region, tending to a curve several decibels away from the BPSK case, and one floor region, whose value becomes lower with increasing M, thus establishing a tradeoff between decoding complexity and desired BER. The reason for this floor has to do with the reference samples y_{n+M-1} lying outside the interval $[0,1]$ as we will show in the sequel. The samples can be correctly decoded with higher probability as the number of consecutive samples taken into account grows, so that there is more information available to decide over the symbol and the redundancy is exploited more effectively.

To show this, we have calculated the probability of error when $M=2$ symbols (for higher M, the theoretical analysis becomes in practice unfeasible). We can consider basically three cases

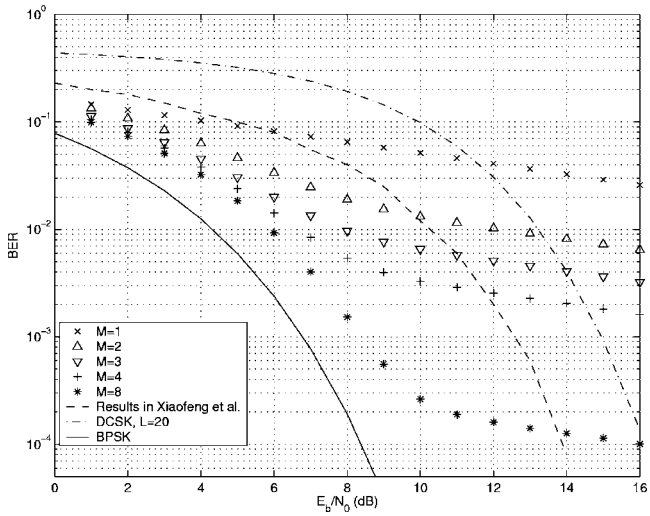

FIG. 4. BER against E_b/N_0 for the Bernoulli shift map in the case of *heuristic decoding*, with $D=20$ bits and $M=1,2,3,4,8$ symbols. The performance of BPSK (Ref. 21), DCSK (Ref. 2) with $L=20$ and other recent results (Ref. 25) are shown for comparison.

- $y_{n+1}<0$
  In this case, as $z_n^0=0$ and $z_n^1=1/2$, we get

$$p_{e1} = P\left[y_{n+1}<0,(z_n^1-y_n)^2<(z_n^0-y_n)^2,x_n<\tfrac{1}{2}\right] + P\left[y_{n+1}<0,(z_n^0-y_n)^2<(z_n^1-y_n)^2,x_n\geqslant\tfrac{1}{2}\right]. \tag{29}$$

- $y_{n+1}>1$
  In this case, $z_n^0=1/2$ and $z_n^1=1$ and the decoded symbol will always be a 1, so

$$p_{e2} = P\left(y_{n+1}>1,x_n<\tfrac{1}{2}\right). \tag{30}$$

- $0\leqslant y_{n+1}\leqslant 1$
  Now, no restriction or normalization is applied, and

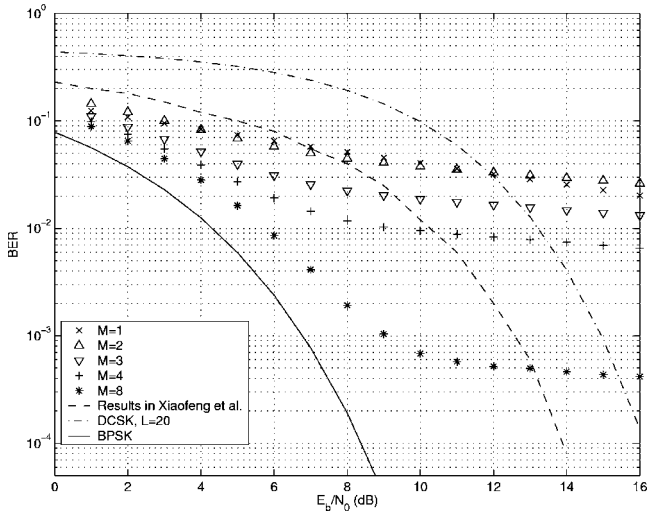

FIG. 5. BER against E_b/N_0 for the modified logistic map in the case of *heuristic decoding*, with $D=20$ bits and $M=1$, 2, 3, 4, 8 symbols. The performance of BPSK (Ref. 21), DCSK (Ref. 2) with $L=20$ and other recent results (Ref. 25) are shown for comparison.

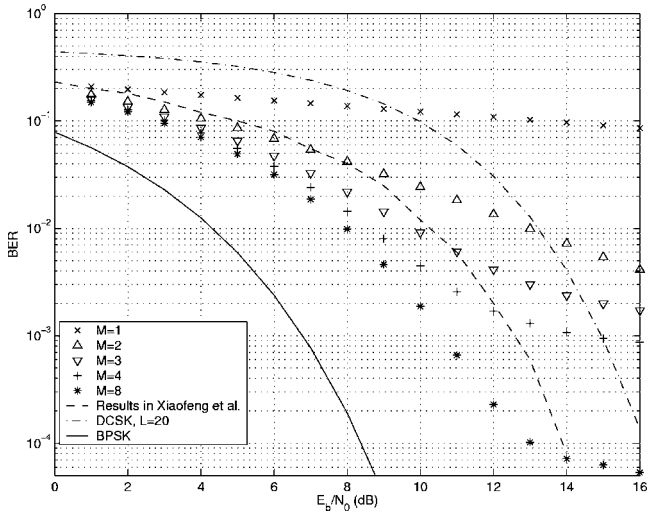

FIG. 6. BER against E_b/N_0 for the FLP map in the case of *heuristic decoding*, with $D=20$ bits and $M=1$, 2, 3, 4, 8 symbols. The performance of BPSK (Ref. 21), DCSK (Ref. 2) with $L=20$ and other recent results (Ref. 25) are shown for comparison.

$$p_{e3} = P\left[0 \leqslant y_{n+1} \leqslant 1, (z_n^1 - y_n)^2 < (z_n^0 - y_n)^2, x_n < \tfrac{1}{2}\right] + P\left[0 \leqslant y_{n+1} \leqslant 1, (z_n^0 - y_n)^2 < (z_n^1 - y_n)^2, x_n \geqslant \tfrac{1}{2}\right]. \tag{31}$$

Knowing that, in the Bernoulli shift case $y_{n+1}=x_{n+1}+n_{n+1}=f(x_n)+n_{n+1}$, $z_n^0=y_{n+1}/2$, $z_n^1=y_{n+1}/2+1/2$, $y_n=x_n+n_n$, and that n_n and n_{n+1} as samples of the AWGN process are independent and identically distributed, we arrive at the following result:

- $y_{n+1}<0$

$$p_{e1}^B = \frac{1}{4}\int_0^{1/2} \operatorname{erfc}\left(2x\sqrt{12\frac{E_b}{N_0}}\right)\operatorname{erfc}\left[\left(\frac{1}{4}-x\right)\sqrt{12\frac{E_b}{N_0}}\right]dx + \frac{1}{2}\int_{1/2}^{1} \operatorname{erfc}\left(2x\sqrt{12\frac{E_b}{N_0}}\right)\left\{1-\frac{1}{2}\operatorname{erfc}\left[\left(\frac{1}{4}-x\right)\times\sqrt{12\frac{E_b}{N_0}}\right]\right\}dx. \tag{32}$$

- $y_{n+1}>1$

$$p_{e2}^B = \frac{1}{4}\operatorname{erfc}\left(\sqrt{12\frac{E_b}{N_0}}\right) + \frac{1}{2\sqrt{48\pi\frac{E_b}{N_0}}}\left(1-e^{-12E_b/N_0}\right). \tag{33}$$

- $0\leqslant y_{n+1}\leqslant 1$

$$p_{e3}^B = \frac{1}{\sqrt{48\pi\frac{E_b}{N_0}}}\int_0^1 \left(\frac{1}{2}-\frac{x}{2}\right)\left\{\operatorname{erfc}\left[\left(\frac{1}{4}-\frac{x}{2}\right)\sqrt{12\frac{E_b}{N_0}}\right] + \operatorname{erfc}\left[\left(\frac{1}{4}+\frac{x}{2}\right)\sqrt{12\frac{E_b}{N_0}}\right]\right\}dx. \tag{34}$$

We have also addressed this problem with the modified logistic map, and, taking into account its own $p(x)$ and the expressions for z_n^0 and z_n^1, we arrive at

- $y_{n+1}<0$

$$p_{e1}^M = \frac{1}{2}\int_0^{1/2} \frac{1}{\pi\sqrt{x(1-x)}}\operatorname{erfc}\left[2\left(\frac{1}{4}-x\right)\sqrt{2\frac{E_b}{N_0}}\right]\times\left(1-\frac{1}{2}\operatorname{erfc}\left\{2\left[-4x(1-x)\right]\sqrt{2\frac{E_b}{N_0}}\right\}\right)dx + \int_{1/2}^{1}\frac{1}{\pi\sqrt{x(1-x)}}\left\{1-\frac{1}{2}\operatorname{erfc}\left[2\left(\frac{1}{4}-x\right)\sqrt{2\frac{E_b}{N_0}}\right]\right\}\times\left(1-\frac{1}{2}\operatorname{erfc}\left\{2\left[-4x(1-x)\right]\sqrt{2\frac{E_b}{N_0}}\right\}\right)dx. \tag{35}$$

- $y_{n+1}<1$

$$p_{e2}^M = \frac{1}{2}\int_0^{1/2}\frac{1}{\pi\sqrt{x(1-x)}}\operatorname{erfc}\left\{2\left[1-4x(1-x)\right]\sqrt{2\frac{E_b}{N_0}}\right\}dx. \tag{36}$$

- $0\leqslant y_{n+1}\leqslant 1$

$$p_{e3}^M = \int_0^{1/2} p(x)\int_{-h(x)}^{1-h(x)}\sqrt{\frac{2}{\pi}\frac{E_b}{N_0}}e^{8E_b/N_0 z^2}\operatorname{erfc}\left\{\left[1+\frac{1}{2}\sqrt{h(x)+z}-\frac{1}{2}\sqrt{1-h(x)-z}-x\right]\sqrt{2\frac{E_b}{N_0}}\right\}dzdx + \int_{1/2}^{1}p(x)\int_{-1+h(x)}^{h(x)}\sqrt{\frac{2}{\pi}\frac{E_b}{N_0}}e^{8E_b/N_0 z^2}\times\left(2-\operatorname{erfc}\left\{\left[1+\frac{1}{2}\sqrt{h(x)+z}-\frac{1}{2}\sqrt{1-h(x)-z}-x\right]\times\sqrt{2\frac{E_b}{N_0}}\right\}\right)dzdx, \tag{37}$$

$$p(x) = \frac{1}{\pi\sqrt{x(1-x)}}, \quad h(x) = 4x(1-x).$$

All the equations, with the exception of Eq. (30), have to be integrated by numerical integration. The results, along with the total bit error probability $p_e=p_{e1}+p_{e2}+p_{e3}$ and the simulation result for $M=2$, are depicted in Figs. 7 and 8.

As it may be seen, the Eqs. (30) and (36) give reason of the behavior of the system as $E_b/N_0\to\infty$, and are responsible, due to their polynomial dependence on E_b/N_0, of the residual BER that may be expected for large signal to noise ratios. The reason for this is that $y_{n+1}>1$ leads to inverse values 1/2 and 1 as stated, both values decoded as $\hat{b}_n=1$ and this will always be decoded on error if $x_n<1/2$ was sent. The other terms (29), (35) affect the performance only in the region of lower E_b/N_0 and their effect is very limited: $y_{n+1}<0$ leads to the inverse values 1/2 and 0, and so, for limited noise, if $x_n<1/2$ and $y_{n+1}\cong 2x_n+n_{n+1}<0$ (in the Bernoulli shift map case), it is most probable that x_n is near 0, $(z_n^0-y_n)^2$ is also near 0 and the symbol is decoded successfully. The terms (31), (37) are dominant in the range of middle

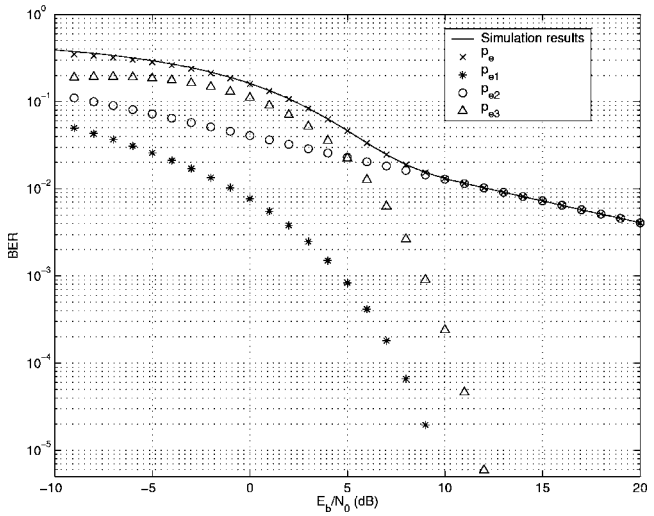

FIG. 7. Analytical probability of error and simulation results in terms of BER against E_b/N_0 for the Bernoulli shift map when performing *heuristic decoding* and $M=2$.

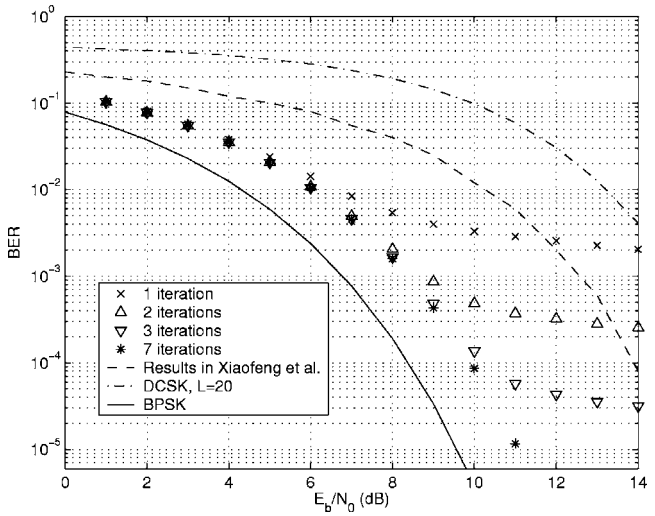

FIG. 9. BER against E_b/N_0 for the Bernoulli shift map in the case of *recursive heuristic decoding*, with $D=20$ bits, $M=4$ symbols and 1, 2, 3, 7 iterations. The performance of BPSK (Ref. 21), DCSK (Ref. 2) with $L=20$ and other recent results (Ref. 25) are shown for comparison.

E_b/N_0, and also drop dramatically as the E_b/N_0 grows, implying that, for a decreasing noise power, the decoding process is able to decode the symbol properly. Therefore, we can say that the increase in decoding complexity by increasing M can compensate for the errors induced when the reference symbol lies outside the interval $[0,1]$, thus reducing the level of the error floor slope found in the simulations. This shows the importance of involving as much samples as possible in the decoding of each sample.

In Figs. 9–11, we can see the results obtained when using the recursive heuristic scheme. In this case, with a fixed complexity ($D=20$, $M=4$), we obtain better results than in the former case: the floor region falls down very fast with each iteration, achieving the bounding curve several decibels away from the BPSK BER in seven iterations, at least to a simulated BER of 10^{-5}. This is a great improvement, since the complexity of the decoder changes from $O(2^M)$ to $O(q2^M)$, with q meaning the number of iterations, and the recursive algorithm performs well with M taking a moderate value. The reason for this better behavior is the propagation of the evidence contained in the whole sequence after each iteration.

In the case of Viterbi decoding (Fig. 12), we can see that, for $P=32$, $L=10$, the results are practically the same as those obtained in the former case with seven iterations (i.e., without floor region down to 10^{-5}). Though the results are not shown, no significant improvement is achieved by increasing P or L, and a slight degradation appears when reducing P. The first and second algorithms achieve the same limit performance with increasing complexity, but the Viterbi decoding converges to the same performance automatically from a given complexity and on. The BCJR decoding (Fig. 13) leads

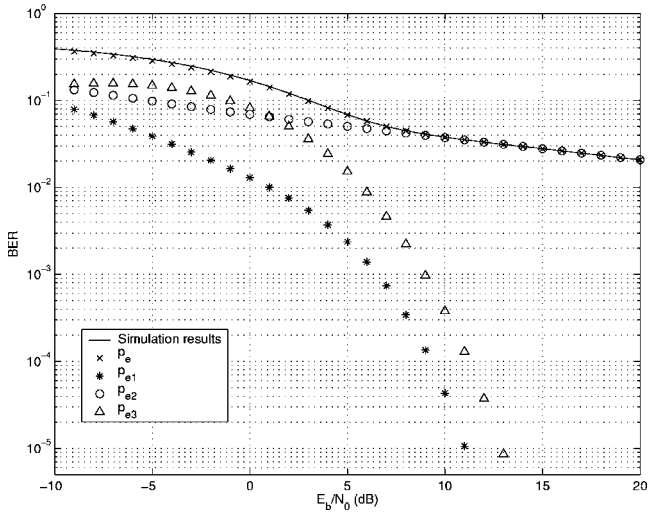

FIG. 8. Analytical probability of error and simulation results in terms of BER against E_b/N_0 for the modified logistic map when performing *heuristic decoding* and $M=2$.

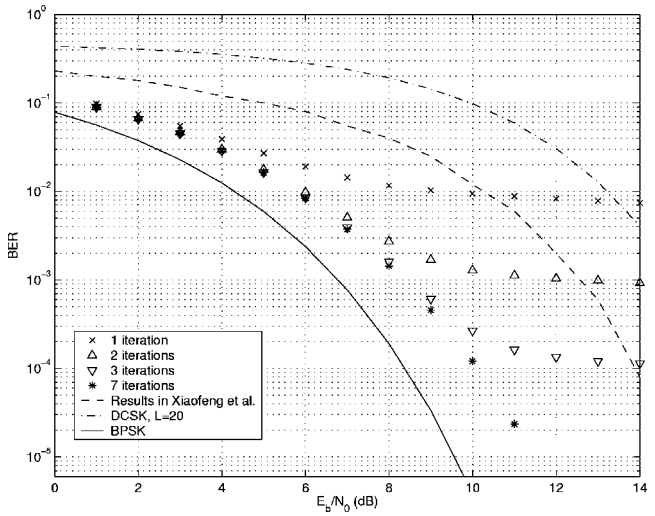

FIG. 10. BER against E_b/N_0 for the modified logistic map in the case of *recursive heuristic decoding*, with $D=20$ bits, $M=4$ symbols and 1, 2, 3, 7 iterations. The performance of BPSK (Ref. 21), DCSK (Ref. 2) with $L=20$ and other recent results (Ref. 25) are shown for comparison.

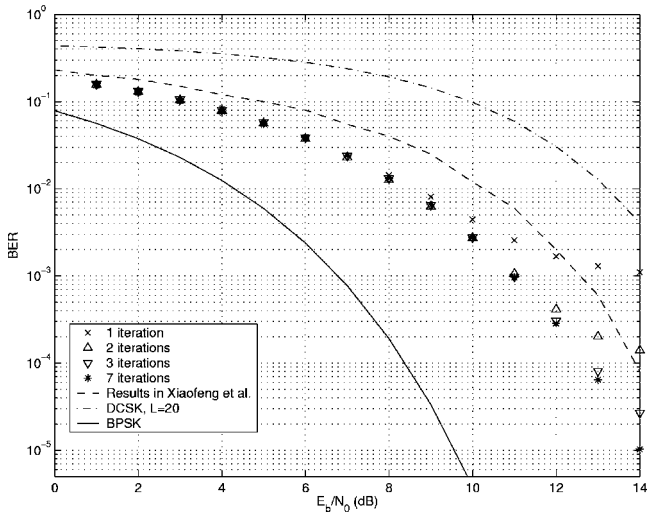

FIG. 11. BER against E_b/N_0 for the FLP map in the case of *recursive heuristic decoding*, with $D=20$ bits, $M=4$ symbols and 1, 2, 3, 7 iterations. The performance of BPSK (Ref. 21), DCSK (Ref. 2) with $L=20$ and other recent results (Ref. 25) are shown for comparison.

to an even closer performance to the BPSK case, practically coinciding with the expected limit performance[9,19] calculated through a minimum distance analysis for the Bernoulli shift map case. In fact, the algorithm proposed is near to the optimal one in terms of BER for the type of encoding used and sheds light on how to look into improving the performance of other encoding schemes recently proposed.[9,12,19] The complexity of the Viterbi and BCJR algorithms grows as $O(LP)$, with P being a power of 2, so that it exhibits the same behavior as the recursive heuristic algorithm, with a linear and an exponential term, another clear advantage against the poor performing heuristic decoding.

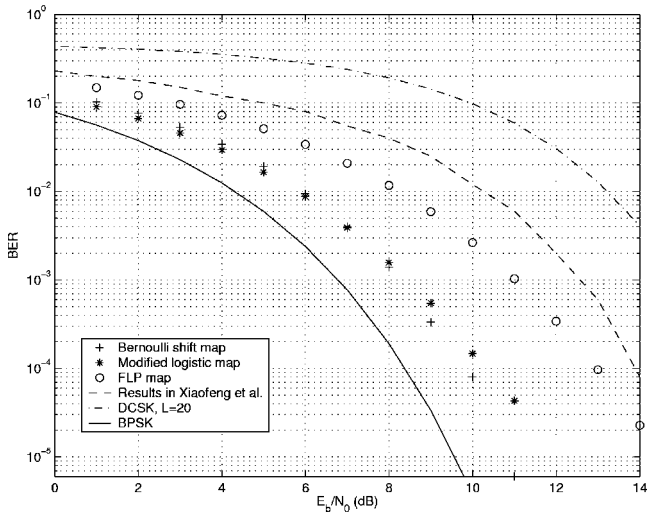

FIG. 12. BER against E_b/N_0 for the Bernoulli shift, the modified logistic and the FLP maps, in the case of *Viterbi decoding*, with $P=32$ states and $L=10$ symbols. The performance of BPSK (Ref. 21), DCSK (Ref. 2) with $L=20$ and other recent results (Ref. 25) are shown for comparison.

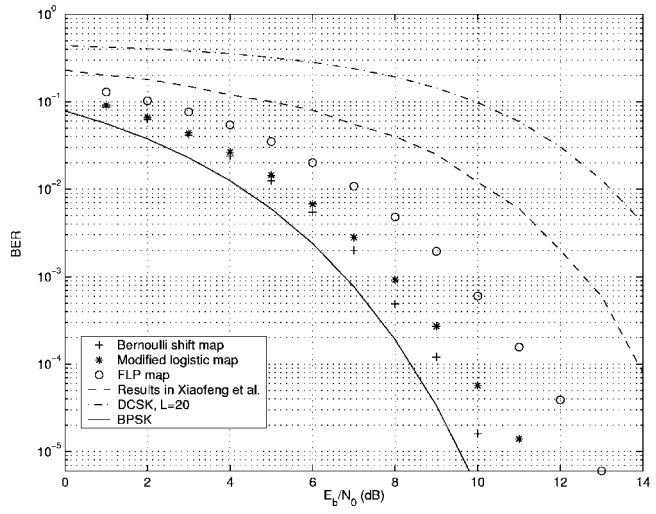

FIG. 13. BER performance for the Bernoulli shift, the modified logistic and the FLP maps, in the case of BCJR *decoding*, with $P=32$ states and $L=10$ symbols. The performance of BPSK (Ref. 21), DCSK (Ref. 2) $L=20$ and other recent results (Ref. 25) are shown for comparison.

V. CONCLUSIONS

In this paper, we have described a communication system based upon chaotic encoding and decoding and we have shown how to use a map designed with a given invariant density (under some constraints) to build a useful transmission scheme by adapting the truncation device already used in Bernoulli shift map schemes to any map conjugated with the former. We have also reviewed four decoding algorithms, through which we have tried to gain insight into the behavior of the system and its performance.

As we have seen, though the described channel encoding and decoding algorithms do not reach at least the BPSK performance, they suppose an improvement of both classical (like DCSK)[26] and recently proposed systems,[25] which are shown in all figures for comparison. Moreover, we have shown two important results:

(1) The received samples which, after the addition of the noise, lie outside the definition interval play a major role in the final BER performance, and
(2) good results are only attainable if the evidence stored in the whole sequence is used and propagated somehow in the decoding process, thus helping to minimize the effect of the outlying samples.

It is worth noting that two of the systems (the one with the Bernoulli shift map and the one with the modified logistic map), lead almost to the same BER, with a slight advantage for the Bernoulli shift map. Both systems were used in order to verify how the map dynamics and the resulting invariant density could affect the BER. Though the pdf achieved with the Bernoulli shift map concentrates more data in the region of the deciding threshold, this does not degrade the performance with respect to the modified logistic map. This means that the implicit dynamics of the maps involved is more important in some cases than the pdf of the encoded sequence. This could be seen in the case of the Viterbi and BCJR decoding in the following way: with the Bernoulli shift map, as

the algorithms proceed forward and backward, the intervals I_i visited maintain a regular spreading (in the forward case, I_i is mapped into to intervals as given by the transition matrix **T**) and despreading ratio (in the backward case, there are two intervals mapped into the interval I_i), thus helping somehow to take advantage of the long distance relationship of the samples. With the modified logistic map, the irregular ratio between intervals [which leads to an irregular distribution of the transition probabilities in the transition matrix following Eq. (28)] seems to hinder the algorithm efficiency. On the contrary, the FLP map has an *a priori* BER (see Fig. 3) worse than both the Bernoulli shift and the modified logistic maps, and this property holds for all the decoding methods studied, meaning that the concentration of data around the threshold point cannot be easily overcome even by increasing the decoding complexity, and this does not seem to happen with the Bernoulli shift map. This means that the pdf can give reason of the overall performance (the FLP map could be seen as a bad candidate for a good BER from the beginning due to the accumulation of data around $x=1/2$), while it does not explain the details of the BER curves (since the modified logistic map could have been seen as a better candidate than the Bernoulli shift map, but in fact the simulations have shown it is not). Some deep research is needed to account for this behavior of the dynamical systems involved, apart from what can be seen through the pdf.

It is to be noticed that the adaptation of the BCJR algorithm is important as a previous step to apply the iterative decoding philosophy of the so-called turbo codes[27] in systems involving one or more than one chaos-channel encoders as the one shown here. We expect that this could boost the performance in terms of BER and make this kind of systems comparable to the ones used in practical communication systems.

Finally, another point worth studying would be looking further into the analytical performance of the systems presented here, and try to generalize the results to a broad kind of maps, for example those lying under the category of the *general antisymmetric maps*

$$x_{n+1}=f(x_n)=\begin{cases}A\left(1-\left|1-2x_n\right|^{\alpha}\right) & \text{if } x_n\leqslant\frac{1}{2}\\ A\left|1-2x_n\right|^{\alpha} & \text{if } x_n>\frac{1}{2},\end{cases} \tag{38}$$

of which the Bernoulli shift map and the modified logistic map are particular examples (with $A=1$, $\alpha=1$ and $A=1$, $\alpha=2$, respectively). This could lead to more refined design procedures to build *ad hoc* maps starting from a desired performance, and not only from a desired invariant density.

ACKNOWLEDGMENTS

The authors acknowledge financial support from the Spanish Ministry of Science and Technology, Project No. BFM2003-03081, and from the Research Program of the Universidad Rey Juan Carlos, Project No. PPR-2004-03.

APPENDIX: MAP SYNTHESIS—AN EXAMPLE

For the sake of comparison and to illustrate the procedure of map synthesis, we carry out the design of a map whose final density has the same shape as the modified logistic map, but with the maximum in the threshold point 1/2:

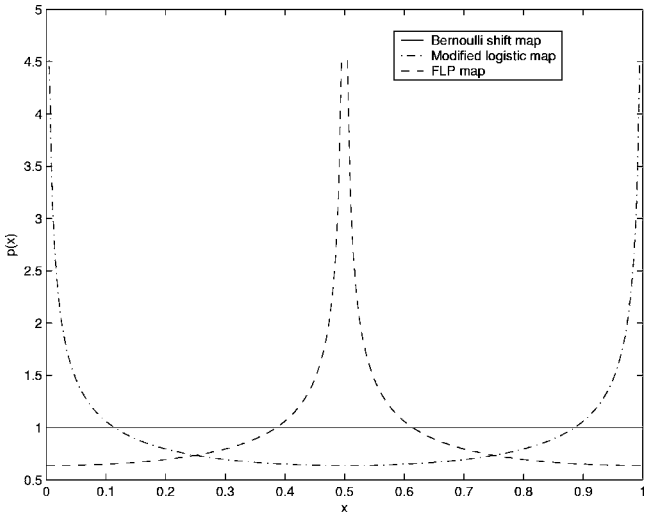

FIG. 14. Invariant probability density functions for the Bernoulli, the modified logistic and the FLP maps.

$$p(x)=\frac{1}{\pi\sqrt{\left(x+\frac{1}{2}\right)\left(\frac{1}{2}-x\right)}},\quad x\in\left[0,\tfrac{1}{2}\right],$$
$$p(x)=\frac{1}{\pi\sqrt{\left(x-\frac{1}{2}\right)\left(\frac{3}{2}-x\right)}},\quad x\in\left(\tfrac{1}{2},1\right]. \tag{A1}$$

This map will also be conjugated of the Bernoulli shift map. All the probability density functions (pdfs) involved are shown in Fig. 14. The map we will obtain through this flipped logistic pdf will be accordingly called *flipped logistic pdf* map (abbreviated in the sequel as FLP map). As mentioned before, the function $g(r)$ which relates the initial condition x_0 to the symbolic state r as defined in Eq. (3) can be obtained starting from the pdf of the map following $g(r)=F^{-1}(r)$. In the case when the map is conjugated to the Bernoulli shift map, if the initial condition for the Bernoulli map is $x_0^B=r$ and the initial condition for the conjugated map is $x_0^C=g(r)$, then all the samples are related through[14]

$$f_C^n(x_0^C)=f_C^n[g(r)]=g[f_B^n(r)]=g[f_B^n(x_0^B)], \tag{A2}$$

where $f_C^n(x)$ is the nth iteration of the conjugated map and $f_B^n(x)$ is the nth iteration of the Bernoulli shift map. The last expression is a direct consequence of the definition of conjugated maps f_1 and f_2, $f_1(x)=\phi\{f_2[\phi^{-1}(x)]\}$, where $\phi(x)$ is a function mapping the interval $[0,1]$ onto itself. The truncated encoding process shown in Eq. (6) makes use of this property (A2).

Therefore, the next task in the design will be to find the expression for $f_C(x)$ starting from the pdf and the relationship (A2). Since $g(r)=F^{-1}(r)$ and the pdf is symmetric with respect to the point 1/2, $g(r)$ has the following properties:

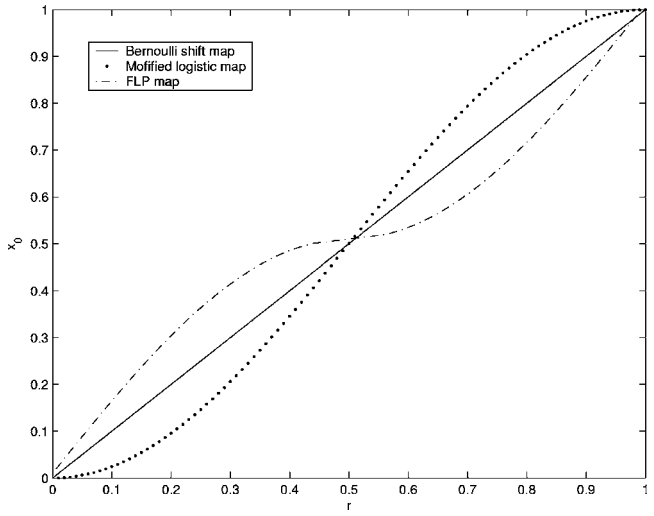

FIG. 15. The different conjugating functions $g(x)$ for the Bernoulli shift, the modified logistic and the FLP maps.

$$g(0)=0, \quad g\left(\tfrac{1}{2}\right)=\tfrac{1}{2}, \quad g(1)=1,$$

$$g\left(\left[0,\tfrac{1}{2}\right]\right)=\left[0,\tfrac{1}{2}\right], \quad g\left(\left[\tfrac{1}{2},1\right]\right)=\left[\tfrac{1}{2},1\right],$$

with $g(r)$ being a monotonously growing function. Taking into account the relationship (A2) and the form of the Bernoulli map, the new map $f_C(x)$ will also have the following property:

$$f_C[g(r)]=\begin{cases} g(2r) & \text{if } r\leqslant \tfrac{1}{2} \\ g(2r-1) & \text{if } r>\tfrac{1}{2}. \end{cases} \tag{A3}$$

As $g(r)$ is invertible, taking $g(r)=x$ and $r=g^{-1}(x)$, we arrive at

$$f_C(x)=\begin{cases} g[2g^{-1}(x)] & \text{if } x\leqslant \tfrac{1}{2} \\ g[2g^{-1}(x)-1] & \text{if } x>\tfrac{1}{2}, \end{cases} \tag{A4}$$

which is the general expression for the new map, provided that its pdf lies in the interval $[0,1]$ and is symmetric with respect to the point $1/2$.

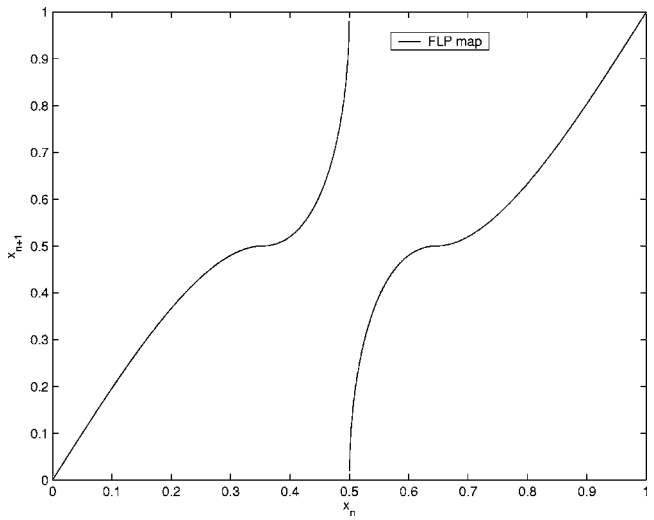

FIG. 16. The figure shows the FLP map.

After some algebra, we can calculate $F(x)$, $g(r)$ and the map for the pdf defined in (A1). The corresponding expressions or the map are

$$F(x)=\begin{cases} \dfrac{1}{2}-\dfrac{2}{\pi}a\cos\left(\sqrt{x+\dfrac{1}{2}}\right) & \text{if } x\leqslant \dfrac{1}{2} \\ \dfrac{3}{2}-\dfrac{2}{\pi}a\cos\left(\sqrt{x-\dfrac{1}{2}}\right) & \text{if } x>\dfrac{1}{2}, \end{cases} \tag{A5}$$

$$g(r)=\begin{cases} \cos^2\left[\dfrac{\pi}{2}\left(\dfrac{1}{2}-r\right)\right]-\dfrac{1}{2} & \text{if } x\leqslant \dfrac{1}{2} \\ \cos^2\left[\dfrac{\pi}{2}\left(\dfrac{3}{2}-r\right)\right]+\dfrac{1}{2} & \text{if } x>\dfrac{1}{2}, \end{cases} \tag{A6}$$

$$f(x_n)=\begin{cases} 4x_n\sqrt{\dfrac{1}{4}-x_n^2} & \text{if } 0\leqslant x_n<\dfrac{1}{2\sqrt{2}} \\ 1-4x_n\sqrt{\dfrac{1}{4}-x_n^2} & \text{if } \dfrac{1}{2\sqrt{2}}\leqslant x_n<\dfrac{1}{2} \\ 4(1-x_n)\sqrt{\dfrac{1}{4}-(1-x_n)^2} & \text{if } \dfrac{1}{2}\leqslant x_n<1-\dfrac{1}{2\sqrt{2}} \\ 1-4(1-x_n)\sqrt{\dfrac{1}{4}-(1-x_n)^2} & \text{if } 1-\dfrac{1}{2\sqrt{2}}\leqslant x_n\leqslant 1. \end{cases} \tag{A7}$$

The different $g(r)$ functions for the three maps involved are presented in Fig. 15, and the FLP map is depicted in Fig. 16. We could now use this three maps to encode a binary sequence using the relationships (6) and see how efficient the resulting encoding systems are in terms of their bit error rate. It is to be noticed that, for the pure encoding process, only $g(r)$ is needed, but the resulting map $f(x)$ will be required on the decoder side.

[1] S. Hayes, C. Grebogi, and E. Ott, Phys. Rev. Lett. **70**, 3031 (1993); **70**, 3034 (1993).
[2] A. Abel and W. Schwarz, Proc. IEEE **90**, 691 (2002).
[3] A. S. Dimitriev, M. Hasler, A. I. Panas, and K. V. Zakharchenko, Nonlinear Phenom. Complex Syst. (Dordrecht, Neth.) **6**, 488 (2003).
[4] G. Kolumban and M. P. Kennedy, IEEE Trans. Circuits Syst., I: Fundam. Theory Appl. **47**, 1673 (2000).
[5] M. P. Kennedy, G. Kolumban, and Z. Jako, IEEE Trans. Circuits Syst., I: Fundam. Theory Appl. **47**, 1702 (2000).
[6] R. Rovatti, G. Mazzini, and G. Setti, IEEE Trans. Circuits Syst., I: Fundam. Theory Appl. **48**, 818 (2001).
[7] G. Kolumban, M. P. Kennedy, Z. Jako, and G. Kis, Proc. IEEE **90**, 711 (2002).
[8] G. Setti, G. Mazzini, R. Rovatti, and S. Callegari, Proc. IEEE **90**, 662 (2002).
[9] S. Kozic, K. Oshima, and T. Schimming, *Circuit Theory and Design, ECCTD 2003, European Conference*, Krakow, Poland, 2003.
[10] T. Schimming and M. Hasler, *Circuits and Systems 2003, ISCAS 2003, IEEE International Symposium*, Bangkok, Thailand, 2003.
[11] G. Setti, R. Rovatti, and G. Mazzini, IEEE Commun. Lett. **8**, 416 (2004).
[12] S. Kozic and T. Schimming, *Circuits and Systems 2005, ISCAS 2005, IEEE International Symposium*, Cobe, Japan, 2005.
[13] A. Baranovsky and D. Daems, Int. J. Bifurcation Chaos Appl. Sci. Eng. **5**, 1585 (1995).
[14] M. P. Kennedy, R. Rovatti, and G. Setti, *Chaotic Electronics in Telecommunications* (CRC Press, Boca Raton, 2000).
[15] J. C. Sprott, *Chaos and Time-Series Analysis* (Oxford University Press,

Oxford, 2003).
[16]J. Schweizer and T. Schimming, IEEE Trans. Circuits Syst., I: Fundam. Theory Appl. **48**, 1269 (2001).
[17]J. Schweizer and T. Schimming, IEEE Trans. Circuits Syst., I: Fundam. Theory Appl. **48**, 1283 (2001).
[18]M. S. Baptista and L. López, Phys. Rev. E **65**, 055201 (2002).
[19]S. Kozic, K. Oshima, and T. Schimming, *Nonlinear Dynamics of Electronic Systems 2003, NDES 2003, IEEE International Workshop*, Scuol, Switzerland, 2003, pp. 141–144.
[20]B. Chen and G. W. Wornell, IEEE Trans. Commun. **46**, 881 (1998).
[21]J. G. Proakis, *Digital Communications* (McGraw-Hill, New York, 1995).
[22]A. J. Viterbi, IEEE Trans. Inf. Theory **13**, 260 (1967).
[23]L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, IEEE Trans. Inf. Theory **IT-20**, 284 (1974).
[24]A. Kisel, H. Dedieu, and T. Schimming, IEEE Trans. Commun. **48**, 533 (2001).
[25]G. Xiaofeng, W. Xingang, Z. Meng, and C. H. Lai, Chaos **14**, 358 (2004).
[26]F. C. M. Lau and C. K. Tse, *Chaos-Based Digital Communication Systems* (Springer, Berlin, 2003).
[27]C. Berrou and A. Glavieux, IEEE Trans. Commun. **44**, 1261 (1996).