

Volume-preserving and volume-expanding synchronized chaotic systems

Louis M. Pecora, Thomas L. Carroll, Gregg Johnson, and Doug Mar

Code 6343, Naval Research Laboratory, Washington, D.C. 20375

(Received 10 February 1997)

Many schemes for synchronizing chaotic systems have emerged in recent years. Several of them have been suggested for use in communications. In this paper we pay special attention to communications issues. We show that one can use dynamical systems techniques to produce synchronized, chaotic systems that have features that are desirable in private and secure communications systems. These features are (i) less structure or pattern to the attractor, (ii) very low time correlations, (iii) more than one positive Lyapunov exponent (hyperchaotic), (iv) synchronizable with one transmitted (scalar) signal, (v) rapid synchronization in few time steps, (vi) the ability to add information to the chaotic carrier in a nonlinear fashion, and (vii) easy to make and analyze. We accomplish this by introducing a simple expanding-folding view of n -dimensional maps which leads to volume-preserving and volume-expanding maps in many cases whose trajectories cover a large nonzero volume of phase space and which have the desired properties. [S1063-651X(97)14710-9]

PACS number(s): 05.45.+b, 84.30.-r, 47.52.+j

I. INTRODUCTION

Many early papers on chaotic synchronization suggested that such behavior might be useful for secure or at least private communications [1–15]. All of these schemes involved some type of masking of small information signals by the chaotic carrier or parameter modulation in the drive (transmitter) by a small information signal. Unfortunately, as demonstrated by Short [16] and Pérez and Cerdería [17] these schemes are not secure and really not very private. They are susceptible to prediction techniques for unmasking the information signals.

Other problems plague the use of “standard” chaotic systems. One is that the attractors occupy well-defined subsets in phase space causing the signals to have a high degree of patterning which will aid in their detection. Time correlations may be quite long, which also aid detection and unmasking. Related to time correlations is the spectra of chaotic systems that typically has a “ $1/f$ ” fall-off or other peaks and features. With only a few recent exceptions (see below), all the systems have only one positive Lyapunov exponent, which also aids prediction. Some of the standard chaotic systems are difficult to construct in circuitry. The Lorenz system is a good example of this [10,18]. Some systems also have rather long synchronization times where the response (receiver) goes through a long transient before achieving good synchronous behavior.

If we examine the requirements of communications systems [19] we see that many of the above problems are exactly what secure communications systems try to avoid. The desire is to generate signals and systems that have little or no patterning, short correlation times, flat spectra, rapid synchronization rates, low predictability, ease of design, easy implementation, and a signal mixing technique that makes extracting information difficult by someone intercepting the transmission.

We show that it is possible to achieve many of these features using concepts from dynamics. We develop a simple technique for synthesizing hyperchaotic, volume-preserving and volume-expanding maps which we will label as VP or

VE maps. The geometric view of the systems are that we have several directions of expansion and simply “fold” the trajectory back into a confined region of phase space whenever it leaves. These systems have little or no structure in phase space (there are no attractors) and hence, very little or no patterning. They are also easy to implement in circuitry. A few recent papers have shown that it is possible to synchronize hyperchaotic systems in a drive-response scenario by using only a scalar signal [20–23]. We use the technique of *synchronous substitution* [24] to achieve similar results so that we can communicate with the hyperchaotic, VP, and VE systems. This approach also allows us to “tune” the system so that synchronization rates are very high (often as rapid as a few iteration steps) and correlations times are low (again, a few steps). Finally, we show that one can use other, more complex methods to mix information with chaotic carriers; these methods appear in some cases to be immune to predictive attempts to extract the information. Here we use a version of the exclusive-or (XOR) function and show that synchronization is still possible, but simple, predictive signal extraction is not.

We note that there are some other relevant studies and similar approaches to the one we present here. A recent paper by Xiao, Hu, and Qu [25] has also made an attempt to use dynamical systems in a fashion compatible with the needs of communications. In that work they found that they could develop arrays of coupled maps that had very low time correlations with each other. That addresses an important feature of private communications.

The use of self-synchronizing shift registers in cryptology was introduced with the paper by Savage [26] and is still mentioned in modern cryptographical books [27]. We note that this scheme is not exactly like the one we propose. In Savage’s scheme the information signal is mixed with the keystream. The output from that mixing is the input into the shift register and is simultaneously transmitted to the receiver where it is input to the receiver’s shift register. Thus, the shift registers on both ends are eventually filled with the same bits. This causes the synchronization. This type of synchronization does not depend on a stability requirement and

will work only for exactly transmitted bits. Dynamical synchronization can often tolerate some degradation in transmitted signals and retain a near synchronous state. Obviously, each approach can have advantages. Comparisons of both need to be done in more detail to determine which would be more appropriate for certain circumstances.

II. VOLUME PRESERVING, LINEAR MAPS

A. Geometric view

We approach producing a VP, chaotic map, by first showing how to make a simple n -dimensional chaotic map, then adjusting the parameters so it is VP or nearly so. We use a linear view of the usual quality of chaotic systems: there is stretching (expansion) and then folding back into a region of past dynamical trajectories. We will see that we can relax the constraint of VP and allow volume expanding (VE) to occur with no loss of generality.

Let L be a linear transformation on the n -dimensional space \mathbb{R}^n . We suppose L has at least one eigenvalue of magnitude greater than 1 and at least one eigenvalue of magnitude less than 1 (these will be adjustable). Then L applied to an arbitrary vector in \mathbb{R}^n will cause it to expand in some direction(s) and contract in others. The linear mapping L will define the *expansion function* part of our chaotic map. Thus given a point in our phase space, say $\mathbf{x}(n)$, we get the next point in time by applying L , thus $\mathbf{x}(n+1) = L\mathbf{x}(n)$. Now how about the folding part?

We define a region of phase space (\mathbb{R}^n) around the origin where we want to confine the dynamics. For simplicity we use an n -dimensional box region: a vector \mathbf{x} is in the region if $x_i \in [-k_i, k_i]$ for each $i = 1, \dots, n$. We define a *folding function* F such that when any component x_i of \mathbf{x} goes beyond the region boundaries $[-k_i, k_i]$ for that component, F moves the system point back into the region so that the i th component is again between $-k_i$ and k_i , otherwise F does nothing. We can think of F as being applied componentwise, since the components are folded independently.

For simplicity we choose all the region boundaries to be at the same k_i values, which we just call k . Obviously, other shaped regions can be used as well as other expansion mapping, e.g., nonlinear ones. We chose the above since it would be easier to implement in a circuit and easier to analyze.

Here are some examples of folding functions. The simplest is the modulus function, as in the Bernoulli shift map in one dimension, where $F(\mathbf{x}) = (x_1 \bmod k, x_2 \bmod k, \dots, x_n \bmod k)$. Another linear folding function is the triangular function (much like the tent map), where $F(\mathbf{x}) = (F(x_1), F(x_2), \dots, F(x_n))$ and F is defined as

$$F(x) = (-1)^q (x - 2 \operatorname{sgn}(x)q), \quad (1)$$

where $q = \operatorname{Int}[(|x| + 1)/2k]$ and $\operatorname{Int}[\]$ means integer part of the argument. The function in Eq. (1) just causes the triangular behavior seen in Fig. 1. We can also use trigonometric functions to fold the phase space point back into the desired region: $F(x) = k \sin(\pi x/k)$ (cf. Gershenfeld and Grinstein [28]). In general any function whose values are bounded by $\pm k$ will work, but obviously some may not give desired behavior. We have stayed with periodic functions that are

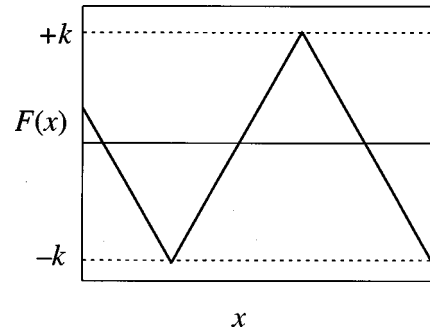


FIG. 1. Triangular folding function.

bounded by $\pm k$. These seem to avoid clustering points in certain areas of phase space and they give the other desirable features we want for communications.

The Lyapunov multipliers will emerge from the ergodic average of the Jacobian of the map. Since the map is linear almost everywhere, we can get a good estimate of those exponents. The Jacobian is given by

$$\mathbf{J} = DF \cdot L \quad (2)$$

In Eq. (2) the factor of DF will be diagonal. For the modulus version of F it is just the identity. For the triangular version we will have ± 1 on the diagonals depending on the “output” components from L . And for the sine version we will have factors of $\cos(x_i)$, where x_i is the value of the i th component after the linear transformation L . For the modulus map the Lyapunov multipliers are just the eigenvalues of L . We can always adjust these so that the map is VP. For the triangular and trigonometric versions we need to calculate the multipliers; however, we find that a good approximation comes from just using the eigenvalues of L .

All of the above follows through if we eliminate the constraint on the eigenvalues to preserve the volume and allow the volume to expand locally.

B. Attaining a stable response: Synchronous substitution

Very often simply applying our original drive-response technique [1] of transmitting one of the drive dynamical variables and substituting it for its counterpart in the response will not lead to a stable driven subsystem. We therefore introduced a form of synchronous substitution [24] to allow us the freedom to transform to variables in which the response will be stable. We show how to do this in general with linear transformations which fit in well with the maps used in this paper and which show the relation of synchronous substitution using linear transformations to standard control theory approaches. We go on to show the stability-variational equation that results in the general case of nonlinear transformations. In the next section we show a particular example which we apply to a circuit to test for the characteristics we desire in a communications system.

The idea of synchronous substitution is simple. We start with the observation that if two systems are in synchrony, then knowing the dynamical variables in one will give the values of the dynamical variables in the other. This is similar to the control theory concept of a state observer. The concept is useful in the case when we can synchronize two systems by sending only one (scalar) variable. This is similar to the

control theory problem of observing a system by using another system [29]. We send only one signal, but we know the values of all variables.

We can make use of this by noting that we can apply an invertible transformation to the (n) drive variables \mathbf{x}^d , say $\mathbf{w} = T(\mathbf{x}^d)$, and transmit w_1 , the first component of \mathbf{w} (we can actually transmit any component, but this is just a reordering). The transmission is just a scalar. On the receiving end with response variables \mathbf{x}^r , if we are near a synchronous state, we can recover one (or more) of the \mathbf{x}^d components by using the inverse T^{-1} to the transformation T even though we have transmitted only one component of \mathbf{w} . This is possible because we are near synchrony, $\mathbf{x}^d \approx \mathbf{x}^r$. The new variables $\mathbf{u} = T(\mathbf{x}^r)$ must be close to \mathbf{w} . Hence, $u_i \approx w_i$ for $i = 1, 2, \dots, n$. We have w_1 from the transmission, so we use u_i or $i = 2, \dots, n$ in place of the remaining w_i to form a new vector $\tilde{\mathbf{w}} = (w_1, u_2, u_3, \dots, u_n) \approx \mathbf{w}$. Then we have at the receiver an estimate $\tilde{\mathbf{x}}^d = T^{-1}(\tilde{\mathbf{w}})$ of the drive variables \mathbf{x}^d which is continually updated by the reception of w_1 . We can now use one or more components of $\tilde{\mathbf{x}}^d$ to drive the response (receiver).

Of course, we are not guaranteed synchronization with any T we choose. However, if we choose to drive the response with any of the $\tilde{\mathbf{x}}^d$ components, we will change the stability of the response since $\tilde{\mathbf{x}}^d$ depends implicitly on \mathbf{x}^r . This means we can use stability of the response as a criterion and adjust T so that we get synchronization. A simpler form of this synchronous substitution was shown in [24] where various linear and nonlinear transformations T were used to stabilize response systems.

In order to simplify our search for good transformations we choose T to be a linear, invertible transformation. This, along with our choice of driving schemes will lead directly to a standard control-theory feedback approach with special cases which yield our original drive-response technique.

We choose to modify the response by adding a feedback term to one or more of the response systems,

$$\mathbf{x}^d(m+1) = F[L\mathbf{x}^d(m)] \quad \text{drive}, \quad (3)$$

$$\mathbf{x}^r(m+1) = F[L\mathbf{x}^r(m) + C(\tilde{\mathbf{x}}^d - \mathbf{x}^r)] \quad \text{response},$$

where C is a coupling matrix. We can rewrite the coupling term as $CT^{-1}(\tilde{\mathbf{w}} - \mathbf{u})$. Now, recall that $\tilde{\mathbf{w}}$ and \mathbf{u} differ only in their first component, hence the remaining components of their difference are zero. Because of this it is easy to show that $(\tilde{\mathbf{w}} - \mathbf{u}) = (\mathbf{K}^T(\mathbf{x}^d - \mathbf{x}^r), 0, 0, \dots, 0)$, where \mathbf{K} is the first row vector of the matrix T . Similarly we can now write the coupling term as $\mathbf{BK}^T(\mathbf{x}^d - \mathbf{x}^r)$, where \mathbf{B} is the first column vector in T^{-1} . We have just derived the standard form for linear feedback control [29]. The form of the coupling in Eq. (3) has been suggested as a control technique for synchronizing chaotic systems [30–32] and has recently been used to synchronize hyperchaotic systems [20]. In our case it is easy to show that this coupling modifies the Jacobian in Eq. (2) to give a new Jacobian,

$$\mathbf{J}' = \mathbf{J} + \mathbf{BK}^T, \quad (4)$$

We show in the next section a specific example of using this approach.

As an aside we note that the control theory approach is a special version of synchronous substitution. We are in general not limited to using linear transformations T , but any invertible function from \mathbb{R}^n to \mathbb{R}^n or even another dynamical system [33] is a valid candidate. In addition we can couple the variables $\tilde{\mathbf{x}}^d$ into the response in many ways other than linear feedback. In the general case we would have a new map, say $\mathbf{G}(\mathbf{x}^r, \tilde{\mathbf{x}}^d)$ dependent on both \mathbf{x}^r and $\tilde{\mathbf{x}}^d$, where the latter is implicitly dependent on \mathbf{x}^r . This leads to a new Jacobian

$$\mathbf{J}' = D_{\mathbf{x}^r}\mathbf{G} + D_{\tilde{\mathbf{x}}^d}\mathbf{G} \cdot D_{\mathbf{w}}T^{-1}\{D_{\mathbf{x}^r}T\}, \quad (5)$$

where $\{D_{\mathbf{x}^r}T\}$ means the Jacobian of the transformation T with the first row replaced by zeroes. The zero row comes about from our replacement of the first component of $T(\mathbf{x}^r)$ with w_1 , the drive variable. For linear, feedback coupling Eq. (5) reduces to Eq. (4) In this paper we adhere to linear transformations, but there most certainly will be cases where nonlinear functions are more appropriate. The reader should see Ref. [24] for examples of these.

Finally, we must admit that although we can optimize the linear stability by our choice of T we have not dealt with the “end points” where the folding takes place. The use of mod or tent functions will occasionally cause loss of synchronization when the drive and response are slightly different (which can occur because of noise or parameter mismatch in analog systems). One system (drive or response) may be folded when the other is not. This will be most severe with the mod function, less so with the tent function. The use of smooth folding functions like the sine will not cause such sudden loss of synchronization. At this point we have no remedy for this problem, except to note that by running our chaotic map system faster than the information signal we can average out the occasional glitches. Other averaging and error-correcting techniques used in spread spectrum communication may also be applicable. We have not yet examined these possibilities. We do note that by optimizing the linear stability we do guarantee rapid resynchronization of the transmitter and receiver, usually in a few steps. Nonetheless, these problems deserve more study.

III. SYSTEM CHARACTERISTICS

We show here that such a VP or VE map system does indeed have many of the properties that we require, namely, nearly homogeneous phase space trajectories (no attractors), very short time correlations, short times to achieve synchronization from arbitrary initial conditions, broad-band, near-white spectra, and some robustness to noise. We use the system described in [34] which we derive here using the above approach.

We choose a specific map given by

$$\left. \begin{aligned} x_{n+1}^d &= ax_n^d + bz_n^d \\ y_{n+1}^d &= cy_n^d + z_n^d \\ z_{n+1}^d &= x_n^d + y_n^d \end{aligned} \right\} \text{mod} 2, \quad (6)$$

where $a < -1$, $b \approx 1$, and $|c| < 1$ are chosen to make the map hyperchaotic. In order to drive our response system we use $\mathbf{B} = (b, 1, 0)$ and $\mathbf{K} = (\Gamma, 0, 1)$ and we have

$$w_n^d = z_n^d + \Gamma x_n^d,$$

$$\tilde{z}_n^d = w_n - \Gamma x_n^r,$$

$$\left. \begin{aligned} x_{n+1}^r &= ax_n^r + b\tilde{z}_n^d \\ y_{n+1}^r &= cy_n^r + \tilde{z}_n^d \\ z_{n+1}^r &= x_n^r + y_n^r \end{aligned} \right\} \text{mod} 2,$$

or

$$\left. \begin{aligned} x_{n+1}^r &= ax_n^r + bz_n^r + b(z_n^d + \Gamma x_n^d - z_n^r - \Gamma x_n^r) \\ y_{n+1}^r &= cy_n^r + z_n^r + (z_n^d + \Gamma x_n^d - z_n^r - \Gamma x_n^r) \\ z_{n+1}^r &= x_n^r + y_n^r \end{aligned} \right\} \text{mod} 2, \quad (7)$$

where we have written everything out to make explicit the dependence of the coupling on the response variables. Using Eq. (7) or Eq. (4) we can calculate the Jacobian of the system and the eigenvalues. It is easy to show that the eigenvalues of the response are $a - b\Gamma$, c , and 0. Since $|c| < 1$ we need only concern ourselves with the first eigenvalue to determine stability. Obviously the condition is that $|a - b\Gamma| < 1$. We see that if $\Gamma = 0$ we will have instability and since $\tilde{z}_n^d = z_n^d$ this case corresponds to trying to use our original drive-response approach which won't work. However, since we have $b \approx 1$ an obvious choice is $\Gamma = a$. This gives a stable response, hence a good candidate for a signal to send is $w_n^d = z_n^d + ax_n^d$. This is what we use in our tests and our circuit.

More details are given on this system in Ref. [24]. Below we also show some results with other folding functions like the triangular and the sine function. We examine the system's characteristics for the particular parameter values $a = -\frac{4}{3}$, $b = 1$, and $c = \frac{1}{3}$. These parameters lead to a VP hyperchaotic system with Lyapunov exponents (0.683, 0.3, -0.986). But similar behavior is also seen for a wide range of parameters that we've investigated $-1 > a > -4$, $c < 1$, and $1 \leq b < 2$.

A. Phase space structures

Figure 2 shows two projections and a three-dimensional (3D) view of the system trajectory ($a = -\frac{4}{3}$, $b = 1$, and $c = \frac{1}{3}$). The behavior is indeed not localized on an attractor, but spread out over much of the space. For these parameters the 3D structure still has some striations [Fig. 2(c)]. Figure 3 was done with the parameters $a = -2$, $b = 1$, and $c = \frac{1}{2}$ and, presumably because of the larger expansion (governed by the parameter a) the phase space structure is more uniform. Our object is not to find the optimum form here, but to show that relatively structureless trajectories are possible with little effort. Similar results ensue with the triangular folding function. The sine folding function retains some phase space structure because of its nonlinearity, namely the trajectory points are most dense near the boundaries and least dense near the origin, although the structure remains spread over most of the phase space region.

B. Time correlations

Figure 4 shows the linear correlation as a function of time shift h calculated simply from

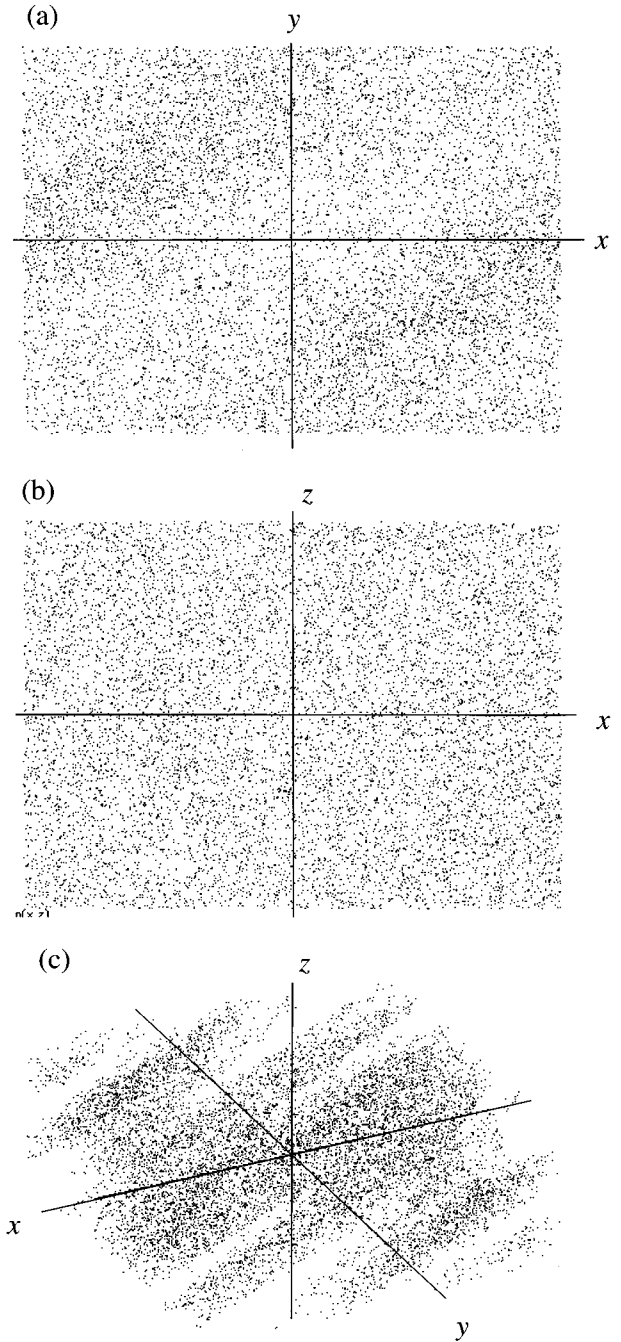


FIG. 2. Trajectories for parameter values ($a = -\frac{4}{3}$, $b = 1$, and $c = \frac{1}{3}$). (a) x - y plane projection of trajectory. (b) x - z plane projection of trajectory. (c) Three-dimensional view of the full phase space trajectory.

$$r(h) = \frac{\frac{1}{N} \sum_i^N [x(i) - m_x][x(i+h) - m_x]}{\sigma_x^2}, \quad (8)$$

where m_x is the mean and σ_x is the standard deviation of the $x(i)$ time series. We see that the correlation falls quickly to zero within one time step. This type of behavior is quite common in the VP systems we have studied. The expansion and folding functions quickly cause the trajectories to lose memory of their initial conditions.

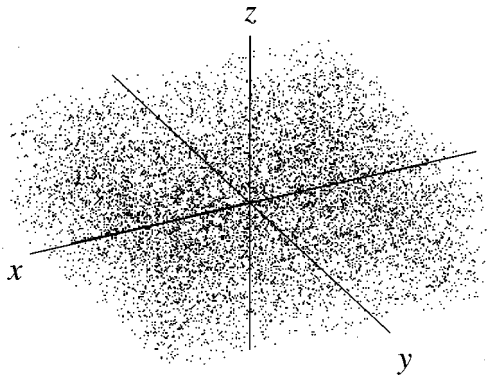


FIG. 3. Three-dimensional view of the full phase space trajectory for parameter values ($a = -2$, $b = 1$, and $c = \frac{1}{3}$).

C. Spectra

Figure 5 shows a typical Fourier amplitude spectrum of the y coordinate for the parameter values ($a = -2$, $b = 1$, and $c = \frac{1}{3}$). The spectrum is broad band and nearly white. We see no discernible features in any of the spectra for this system for a wide range of parameters, although as the “expanding coefficient,” a , is decreased toward 1, the spectra begin to look more like a “ $1/f$ ” type.

D. Time to synchronization

We used the triangular version of the VP and VE maps to test the time to synchronization. We used parameters for VP and set to guarantee x synchronization in one step (see above) ($a = -\frac{4}{3}$, $b = 1$, and $c = \frac{1}{3}$) and x synchronization in several steps ($a = -\frac{4}{3}$, $b = 1.1$, and $c = \frac{1}{3}$) and parameters for VE for immediate x synchronization ($a = -\frac{4}{3}$, $b = 1$, and $c = \frac{1}{3}$) and x synchronization in several steps ($a = -\frac{4}{3}$, $b = 1.1$, and $c = \frac{1}{3}$). All cases were similar, so we report on one (VP). We tracked the decay of the “distance” between the drive and response $\Delta r = \sqrt{(x^r - x^d)^2 + (y^r - y^d)^2 + (z^r - z^d)^2}$. Because the map is, for the most part, linear, we get exponential decay of $\Delta r \sim e^{-\lambda t}$, where $\lambda \approx 1.827$. This gives a decay of a factor of 1000 in about four steps on average. Hence, if we had 10 bits of accuracy in a communications line, we would be down to 1 bit difference between drive and response in four steps even if both were initially different by the maximal amount (2^{10}). Tests for the modulus map give similar results.

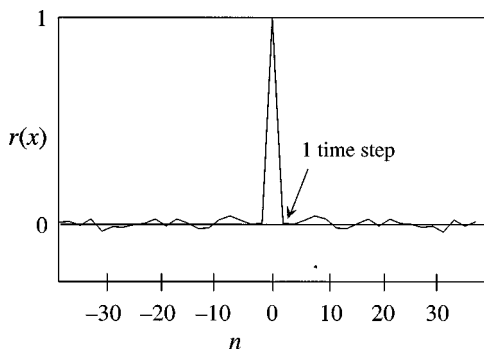


FIG. 4. Linear correlation as a function of iteration step (n) shift for the x variable for parameter values ($a = -\frac{4}{3}$, $b = 1$, and $c = \frac{1}{3}$). The correlation drops to zero after one time step.

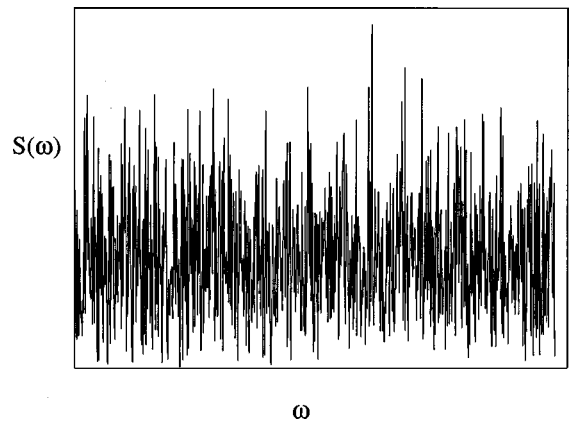


FIG. 5. Fourier amplitude spectrum of the y variable for parameter values ($a = -2$, $b = 1$, and $c = \frac{1}{3}$).

E. Influence of noise

We tested, numerically, the addition of noise on the synchronization process. This presumably also gives an indication of robustness to parameter mismatch, although we did not test that directly. We again used the triangular map with folding function bounds at ± 1 for all components. The parameters were ($a = -\frac{4}{3}$, $b = 1$, and $c = \frac{1}{3}$).

Figure 6 shows the phase plots of x^r versus x^d and y^r versus y^d for 1% and 2% peak amplitude noise levels (rms noise levels correspond to 0.67 and 0.134, respectively) for uniform, bounded random noise added to x^r and y^r simultaneously. We see that the maps are sensitive to noise. The dependence of the amount of synchronization loss, $\langle \Delta y \rangle_{\text{rms}}$ versus the rms noise level η is linear from small noise levels to about 20% noise to signal ratios. The relation is approximately $\langle \Delta y \rangle_{\text{rms}} = 2.9\eta$. That is, for each percent noise we add we throw the system out of synchronization in the y component by a percentage that is almost three times as

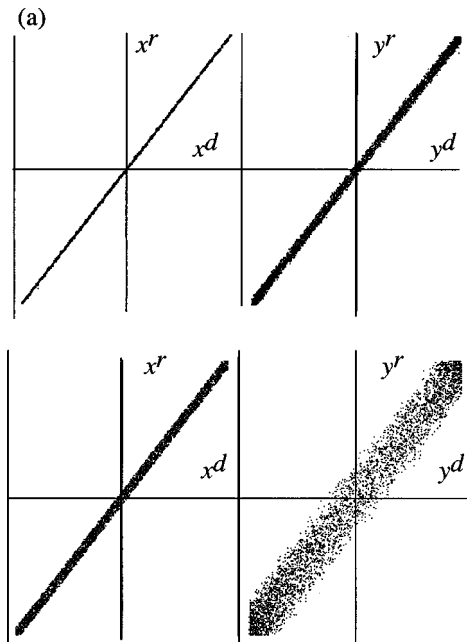


FIG. 6. Phase plots of synchronization between x and y components for the triangular map version for (a) 1% rms noise and (b) 2% rms noise.

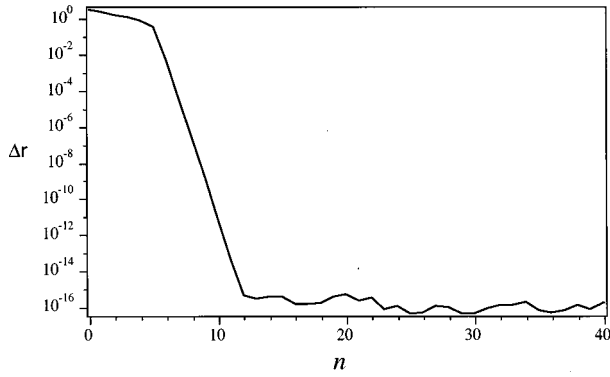


FIG. 7. Plot of average synchronization error versus iteration step (n) for optimized synchronization scheme.

large. The y components show the most sensitivity to noise as can be seen from Fig. 6. Despite the apparent sensitivity to noise in the numerical models these maps can be built in electronic circuits.

F. Other drive-response combinations

In order to apply chaotic synchronization to communications, it is necessary to have many different drive-response circuit pairs. If one puts the drive and response in the form of Eq. (2) as Peng *et al.* [20] do for hyperchaotic systems, we are able to find many stable response systems. Peng *et al.* found stable response systems by estimating what parameter combinations would lead to stability and then searching near these parameters: we were able to automate this process to find many response systems.

We are able to generate other drive-response combinations by using the techniques of Peng *et al.* [20] as shown in Eqs. (3) and (4). For the map of Eq. (6) with $a = -\frac{4}{3}$, $b = 1$, and $c = \frac{1}{3}$, the Jacobian is constant. To apply the method of Peng *et al.*, we must choose three components for the vector \mathbf{B} and three components for the vector \mathbf{K} . We used existing numerical routines for optimization [35] to find the \mathbf{B} and \mathbf{K} vectors that minimized the modulus of the largest Lyapunov exponent of the response map. Starting from an arbitrary point in the six-dimensional \mathbf{BK}^T parameter space, we determined the largest value of the magnitude of the eigenvalues of the response Jacobian, which we shall call $|\mu_{\max}|$. We then allowed the components of \mathbf{B} and \mathbf{K} to be varied in a manner that seeks out the nearest local minimum of $|\mu_{\max}|$. Repeating the procedure with a dense array of starting points revealed dozens of combinations that reduced $|\mu_{\max}|$ to zero, even though we limited the range of \mathbf{B} and \mathbf{K} to ensure small feedback signals. The optimized combinations can significantly enhance the convergence of the response system to the drive. In Fig. 7 we show the vector difference between the two systems as the coupling is initiated at $t=0$. Here $\mathbf{K}=(1.559,1.030,0.6531)^T$ and $\mathbf{B}=(-0.9067,-0.1176,0.8645)^T$ corresponding to $\mu_{\max}=9.8 \times 10^{-3}$.

With multiple combinations of \mathbf{B} and \mathbf{K} vectors that ensure synchronization, one could conceivably design a transmitter that could be tuned to exclusively synchronize a certain receiver with fixed coupling parameters. The components of \mathbf{K} at the receiver constitute a “synchronization address” in the sense that the only receiver that will

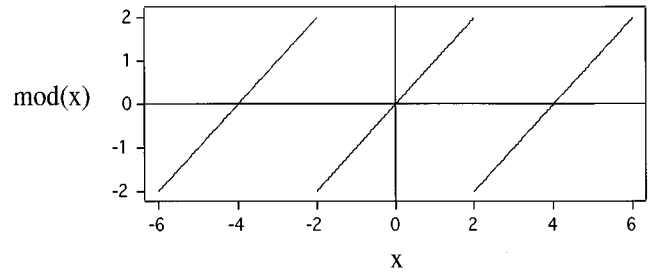


FIG. 8. Modulus function used for the map of Eq. (7).

synchronize to the transmitter is the one with the same \mathbf{K} . With multiple receivers, each with unique parameters, the transmitter’s \mathbf{K} could be tuned to switch from one specific receiver to another. For example, if a second response system was built with $\mathbf{K}=(-1.285,-0.7682,-0.2000)^T$ and $\mathbf{B}=(0.8957,0.0528,-1.022)^T$, this system would synchronize with the drive system above only when \mathbf{K} of the drive is switched to the same values, at which point the response would converge according to $\mu_{\max}=4.2 \times 10^{-3}$.

IV. VOLUME-PRESERVING CIRCUIT

A. Circuit description

We built a circuit to simulate the map of Eq. (6) with $a = -\frac{4}{3}$, $b = 1$, and $c = \frac{1}{3}$. Our modulus function for the circuit is symmetric about the origin, and we show this modulus function in Fig. 8. Figure 9 is a block diagram of the circuit. The math block in the circuit is actually fairly simple, consisting only of operational amplifier adders and subtractors. The sample and hold blocks of the circuits consist of two cascaded sample and hold amplifiers. The sample and hold amplifiers are alternately clocked, so one amplifier is sampling while the other is holding, making an analog memory. The modulus blocks, which perform the mod function shown in Fig. 8, were the most complicated part of the circuit to build and match because they produce a discontinuous function. The clock for this circuit ran at 6 KHz.

B. Circuit tests

Figure 10 is a plot of y^d versus x^d from the map circuit. The plot fills the phase space, showing no obvious correlation between x^d and y^d . A power spectrum of x^d is shown in Fig. 11. The power spectrum is quite flat. The autocorrelation function of x^d drops to 0 in three clock cycles. Since the

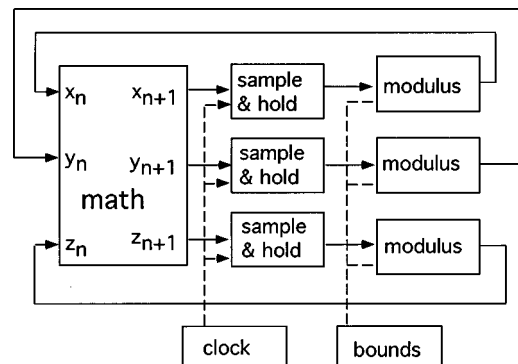


FIG. 9. Block diagram of the map circuit.

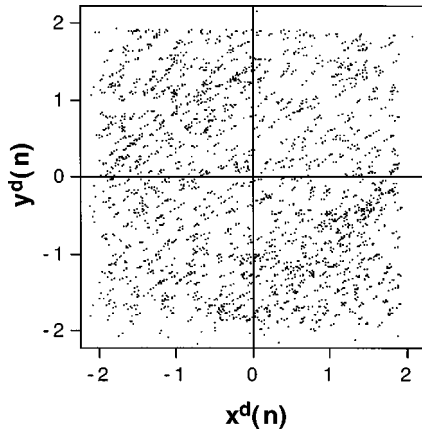


FIG. 10. The y^d signal from the map circuit vs. x^d signal from the map circuit.

output of the circuit changes only once every 2 clock cycles, the autocorrelation function of x^d is almost a delta function.

We also built a response circuit with the same parameters as the drive circuit. Figure 12 shows x^r (from the response circuit) versus x^d (from the drive circuit). The circuits are synchronized most of the time. There are many desynchronized points visible in Fig. 12. The discontinuity in the modulus functions is difficult to match, so that synchronization in the circuits is not perfect.

V. COMMUNICATIONS

In order to test communications possibilities with the VP and VE maps we chose a method for mixing the chaos and information signal that can still preserve perfect synchronization in the case of no noise and no parameter mismatch. This type of signal mixing can be attained by mixing the signal with a dynamical variable and reinjecting that variable into the map on the next time step. This type of mixing was proposed by Volkovkii and Rul'kov [8], Wu and Chua [14], and Kocarev and Parlitz [36]. This is shown schematically in Fig. 13. Equations (6) and (7) are modified as follows:

$$\left. \begin{aligned} x_{n+1}^d &= ax_n^d + bz_n^{*d} \\ y_{n+1}^d &= cy_n^d + z_n^{*d} \\ z_{n+1}^d &= x_n^d + y_n^d \end{aligned} \right\} \text{mod}2, \tag{9}$$

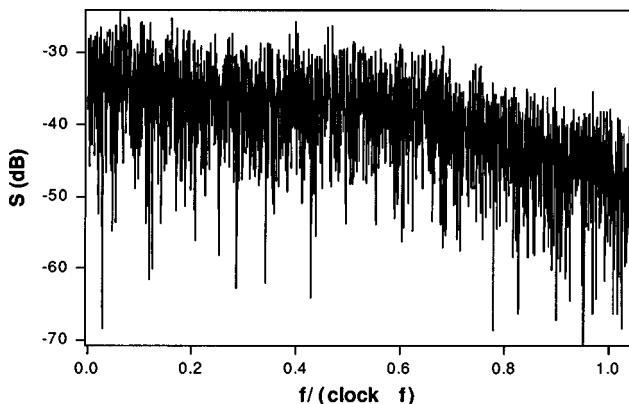


FIG. 11. Power spectrum S of the transmitted signal w from the map circuit with a clock frequency of 6 kHz. The frequency is plotted as a fraction of the clock frequency.

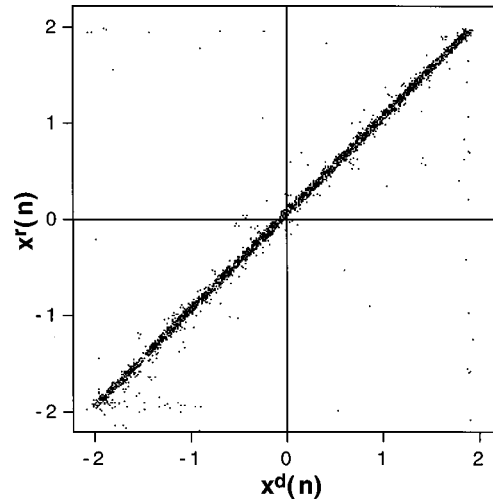


FIG. 12. x^r from the response circuit vs x^d from the drive circuit showing that the response circuit does synchronize to the drive.

$$\left. \begin{aligned} x_{n+1}^r &= ax_n^r + b\tilde{z}_n^{*d} \\ y_{n+1}^r &= cy_n^r + \tilde{z}_n^{*d} \\ z_{n+1}^r &= x_n^r + y_n^r \end{aligned} \right\} \text{mod}2,$$

where z_n^{*d} is the z_n^d dynamical variable with the information mixed in and \tilde{z}_n^{*d} is the z variable extracted from the synchronous substitution from w_n^* , viz.

$$\begin{aligned} z_n^{*d} &= q(z_n^d, i_n), \\ w_n^* &= T(x_n^d, z_n^{*d}) = ax_n^d + z_n^{*d}, \\ \tilde{z}_n^{*d} &= w_n^* - ax_n^r. \end{aligned} \tag{10}$$

In Eq. (10) q is the invertible mixing function. At the receiver when the systems are in synchrony we reproduce $z_{n+1}^d = z_{n+1}^r$ the value of the z component before signal mixing. We save this value until the next time step and use it to recover the information,

$$i_n^r = q^{-1}(z_n^d, \tilde{z}_n^{*d}) \tag{11}$$

and in synchronization $i_n = i_n^r$.

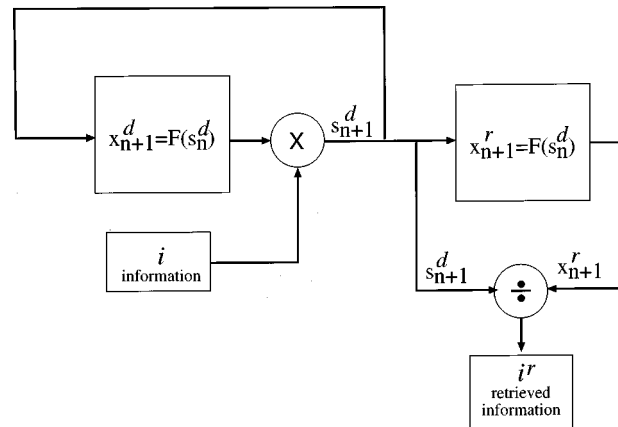


FIG. 13. Block diagram showing the method for mixing the information signal into the drive as part of the system dynamics.

We examine two methods for mixing chaos and information. One is a simple addition of signals known as signal masking [9–11,14,37,38] in which $z^* = z + i$. The other mixing function is one which, to our knowledge, has not been tried before in chaotic circuits, but is common in the communications field, an exclusive-or (XOR) function, in which $z^* = lz \text{ XOR } li$, where l is a normalization factor which determines the number of bits we use to represent the signals. We subject both methods to predictive attempts to extract the information signals [16,17].

A. Chaotic masking and signal extraction

One simple way to send information on a chaotic signal is to add the information signal to one of the dynamical variables in the chaotic system. We added the information signal $i = 0.5 \sin(2\pi f_i t)$ to the second equation in Eqs. (6),

$$y_{n+1} = \left(\frac{1}{3}\right)y_n + z_n + i. \quad (12)$$

We decoded the information signal by looking at the synchronization error Δ in the receiving circuit,

$$\Delta = \bar{z}_n - x'_{n-1} - y'_{n-1} \pmod{2}. \quad (13)$$

We found the signal to noise ratio at the information frequency f_i from the power spectrum of a 16 384 point time series of Δ digitized at 20 kHz. The signal to noise ratio when $f_i = 700$ Hz was 33 dB. The signal to noise ratio at f_i was essentially the same as we varied f_i from 10 Hz to 3 kHz. The information signal f_i was not visible in the power spectrum of the transmitted signal w .

We checked the security of this encoding method by using a predictive algorithm to extract the encoded information signal. The predictive algorithm was based on the work of Short [16]. We embedded the chaotic signal in a phase space using the method of delays and fit local linear maps to the resulting phase space plot. We then used the maps to predict the chaotic time series. The small information signal mixed with the chaos-caused errors in the prediction. We used the Fourier spectrum of the prediction errors to create a filter which we then applied to the chaotic time series to extract the information signal. The method for Short [16] was essentially the same, but Short used more sophisticated fitting. We found that the Short signal extraction method had no trouble extracting the periodic information signal when the signal was simply added to a variable.

B. Nonlinear signal mixing, XOR

The XOR function test is implemented on a triangular map with the variables kept between ± 1 and choosing the level of digitization through the normalization factor l where $q(z, i) = lz \text{ XOR } li$ accomplished by taking the integer parts of lz and li , using a bitwise exclusive-or and rescaling the output by $1/l$. The XOR function is its own inverse and to accomplish this we use the same normalization, integer part, bit-wise exclusive-or, and rescaling by $1/l$. In all our demonstrations here we use $l = 1024$. This means a one-bit error corresponds to approximately a 0.001 absolute error in the signal.

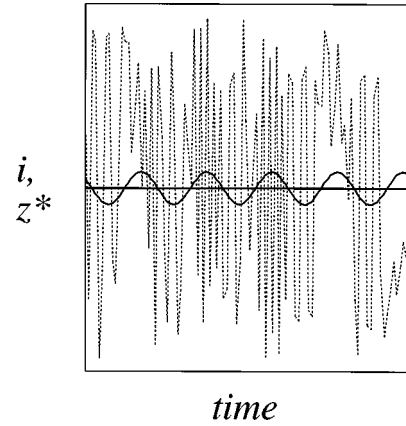


FIG. 14. Sine wave i ($A = 0.1$) extracted from XOR mixed transmitted signal (black line). The simultaneously transmitted XOR signal (z_n^{*d}) is shown in dotted lines in the background.

We ran several tests using $i = A \sin(\omega t)$. The first test was with a small amplitude sine wave, $A = 0.1$ and a frequency of 10π , so the sine wave repeats every 20 points. Figure 14 shows the extracted information signal, i_n^r , at the response. We see a clean extraction as shown by the black line against the background of the transmitted w_n^* shown in grey, although there are occasional “glitches” in the sine wave. These occur because we are using a limited resolution (1024 bits) and the XOR function is nearly discontinuous, hence occasionally we get a 1 bit error that causes a substantial jump in the extracted signal that is *not* there in the original sine wave.

Figure 15 shows the extracted information signal, i_n^r , in the case we have 1% additive noise in the response system, while Fig. 16 show the Fourier amplitude spectrum of the result of the XOR of the z variable and the sine wave. The noise corrupts the sine wave. This is because of the discontinuous nature of the XOR function: small changes can cause large errors for certain z and sine-wave combinations. However, in Fig. 17 we see from the Fourier spectrum of the extracted sine-wave that the single frequency peak remains large (approximately 30 dB above the noise floor) and we

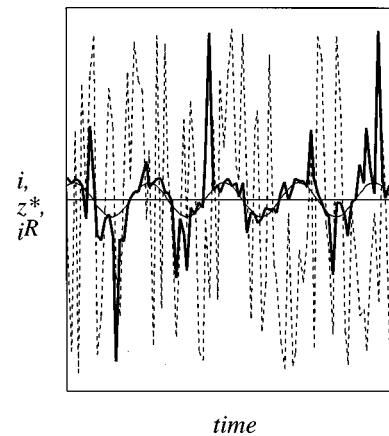


FIG. 15. Similar to Fig. 14 except 1% noise added to the transmitted signal. Sine wave i^r ($A = 0.1$) extracted from XOR mixed transmitted signal (bold black line). The simultaneously transmitted XOR signal (z_n^{*d}) is shown in dotted lines in the background and the original sine wave i in thin solid line.

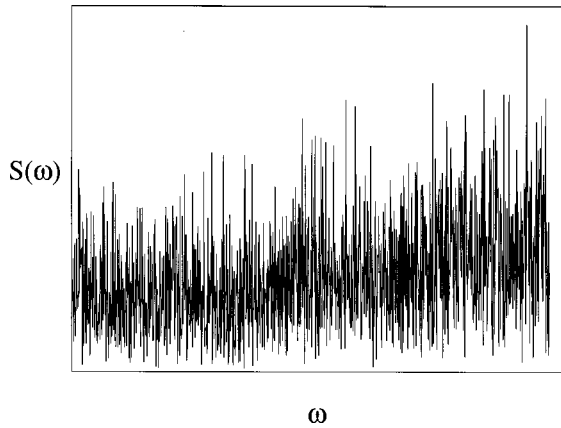


FIG. 16. The Fourier amplitude spectrum of the transmitted signal z_n^{*d} ($A=0.1$) with 1% noise.

could restore much of the sine wave's characteristics with standard signal processing techniques.

In Fig. 18 we see the recovery of a large amplitude sine wave, $A=1.0$, both without and with 1% noise. Figure 19 shows the quality of synchronization when there is 1% noise. The spectrum of the recovered sine wave is shown in Fig. 20 in the case of the added noise. We do some simple signal processing on the recovered signal using a bandpass filter to isolate the main peak. Figure 21 shows the results of this. There is only a small modulation added to the original sine wave.

We used the small ($A=0.1$) and large ($A=1.0$) amplitude sine wave signals to test whether we could extract the signals from the transmitted signal z_n^{*d} using the usual predictive strategies. When the sine wave was small, the information signal did not greatly change the chaotic signal. The Short [16] predictive algorithm was able to extract the information signal for $A=0.1$.

When $A=1.0$, then the transmitted signal no longer looked like the original chaotic signal with a small amount of noise. The chaotic signal was greatly altered by the information signal. It was not possible to get a good delay embedding of the original chaotic system, so we could not fit local maps as we could when the information signal was small.

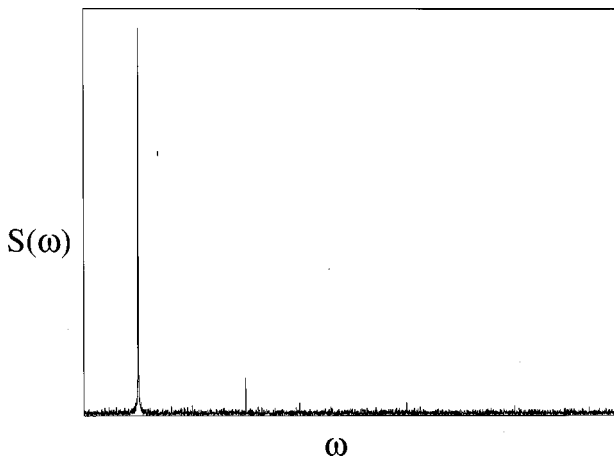


FIG. 17. The Fourier amplitude spectrum of the extracted sine wave ($A=0.1$) with 1% noise. The original peak is clearly present with some higher frequency noise and harmonics.

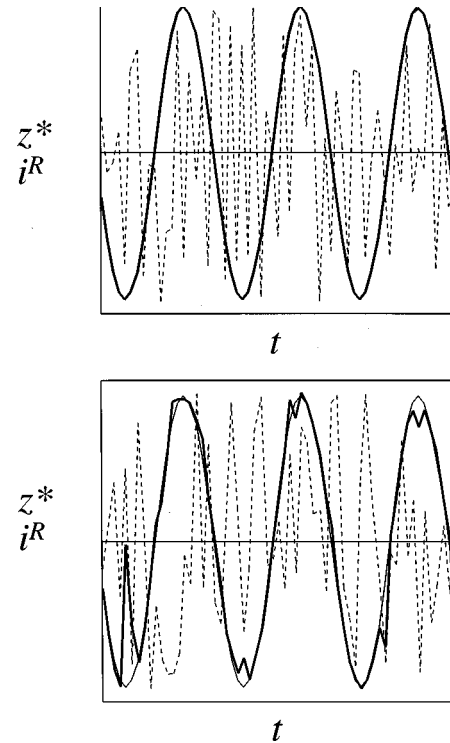


FIG. 18. Sine waves ($A=1.0$) extracted from XOR mixed transmitted signal (black line), (a) with no noise and (b) with 1% noise. The simultaneously transmitted XOR signal (z_n^{*d}), is shown in grey lines in the background.

We were not able to extract the information signal from the chaotic signal using a simple version of the predictive algorithm.

VI. CONCLUSIONS AND REMARKS

The simple stretch and fold approach to linear maps yields synchronizable systems that enjoy characteristics more appropriate for communications than do most chaotic systems that possess an attractor that has a low volume and high degree of patterning in phase space. The map approach should be significantly better since it is discrete and should

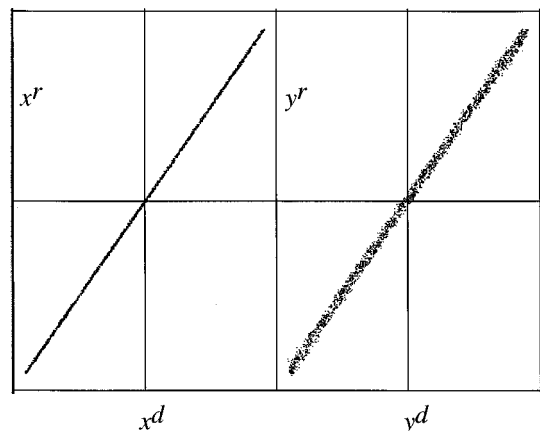


FIG. 19. Synchronization phase plots of the drive and response variables when the XOR mixing function is used and there is 1% additive noise present in the channel.

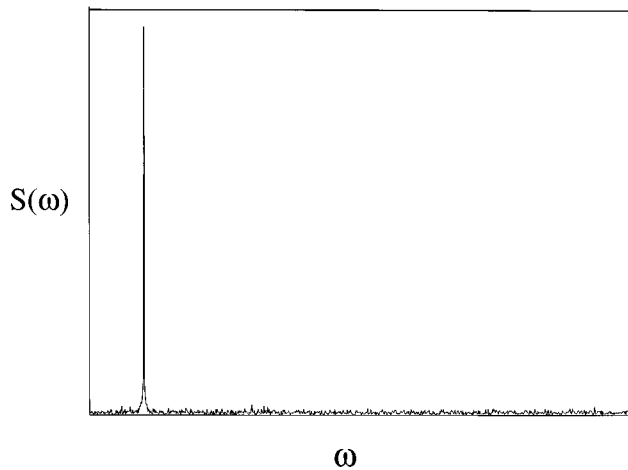


FIG. 20. Fourier amplitude spectrum of the extracted sine wave ($A=1.0$) with 1% noise. The sine peak is obviously much cleaner in here than in Fig. 16.

match better with digital systems, as well as remain easy to design and analyze.

The use of synchronous substitution allows us to tune the transmitted signal and the response so that synchronization is more robust and can occur in a short time period. The use of such synchronous transformations is good from another standpoint. Producing unique chaotic systems en masse is a difficult challenge. However, we can present the same chaotic transmitter in many different guises simply using invertible transformations.

We see by the use of our nonlinear mixing (XOR) that this method is probably preferred to simple chaotic signal masking, although the XOR is not a panacea since there can

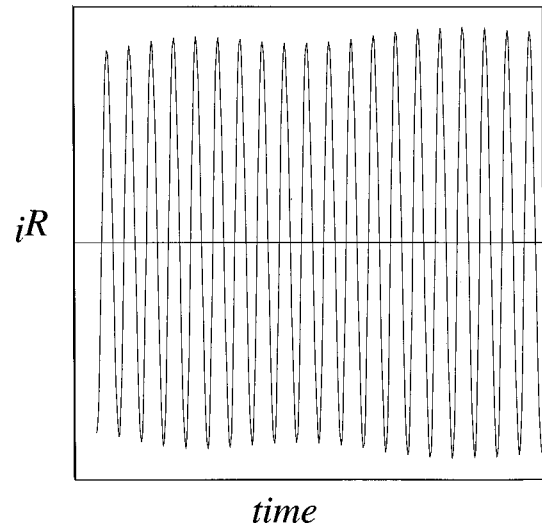


FIG. 21. Resulting sine wave from a bandpass filter applied to the spectrum in Fig. 19 to allow only the main peak to remain.

be other ways to attacks XOR encryption [39]. Whether these attacks, which depend on a finite key length make sense for a chaotic keystream in which there is no key length, apparently, is not clear. Another problem with the XOR is also one of its good features: it has extreme variations in values with only slightly different arguments (it is continuous, but not differentiable). This makes loss of synchronization easier and decreases the robustness of the system in the presence of noise or parameter mismatch. Much more work needs to be done to either find robust maps or to replace the XOR with a highly nonlinear, secure mixing function, but one which will not cause as much trouble in the presence of noise and parameter mismatch.

-
- [1] L. M. Pecora and T. L. Carroll, *Phys. Rev. Lett.* **64**, 821 (1990).
 - [2] Louis M. Pecora and Thomas L. Carroll, *Phys. Rev. A* **44**, 2374 (1991).
 - [3] T. L. Carroll and L. M. Pecora, *Int. J. Bifurcation Chaos Appl. Sci. Eng.* **2**, 659 (1992).
 - [4] T. L. Carroll, *IEEE Trans. Circuits Syst.* **42**, 105 (1995).
 - [5] L. O. Chua *et al.*, *Int. J. Bifurcation Chaos Appl. Sci. Eng.* **2**, 705 (1992).
 - [6] L. O. Chua *et al.*, *Int. J. Bifurcation Chaos Appl. Sci. Eng.* **2**, 973 (1992).
 - [7] Gullicksen, in *Proceedings of the First Experimental Chaos Conference*, edited by M. Spano S. Vohra, M. Shlesinger, L. Pecora, and W. Ditto (World Scientific, Singapore, 1992).
 - [8] A. R. Volkovskii and N. F. Rul'kov, *Sov. Tech. Phys. Lett.* **19**, 97 (1993).
 - [9] Lj. Kocarev *et al.*, *Int. J. Bifurcation Chaos Appl. Sci. Eng.* **2**, 709 (1992).
 - [10] K. Cuomo and A. V. Oppenheim, *Phys. Rev. Lett.* **71**, 65 (1993).
 - [11] K. M. Cuomo, A. V. Oppenheim, and S. H. Strogatz, *IEEE Trans. Circuits Syst.* **40**, 626 (1993).
 - [12] N. F. Rul'kov *et al.*, *Int. J. Bifurcation Chaos Appl. Sci. Eng.* **2**, 669 (1992).
 - [13] N. F. Rulkov and A. R. Volkovskii, *SPIE Proc.* 132 (1993).
 - [14] C. W. Wu and L. O. Chua, *Int. J. Bifurcation Chaos Appl. Sci. Eng.* **3**, 1619 (1993).
 - [15] V. S. Anishchenko, *Int. J. Bifurcation Chaos Appl. Sci. Eng.* **2**, 633 (1992).
 - [16] Kevin M. Short, *Int. J. Bifurcation Chaos Appl. Sci. Eng.* **4**, 959 (1994).
 - [17] Gabriel Pérez and Hilda A. Cerderia, *Phys. Rev. Lett.* **74**, 1970 (1995).
 - [18] K. Cuomo (private communication).
 - [19] Keith Godfrey, *Perturbation Signals for System Identification* (Prentice Hall, New York, 1993).
 - [20] J. H. Peng *et al.*, *Phys. Rev. Lett.* **76**, 904 (1996).
 - [21] L. S. Tsimring and M. M. Sushchik, *Phys. Lett. A* **213**, 155 (1996).
 - [22] L. Kocarev, U. Parlitz, and T. Stojanovski, *Phys. Lett. A* **217**, 280 (1996).
 - [23] A. Tamasevicius *et al.*, *Electron. Lett.* **32**, 1536 (1996).
 - [24] T. L. Carroll, J. F. Heagy, and L. M. Pecora, *Phys. Rev. E* **54**, 4676 (1996).
 - [25] J. H. Xiao, G. Hu, and Z. Qu, *Phys. Rev. Lett.* **77**, 4162 (1996).
 - [26] J. E. Savage, *Bell Syst. Tech. J.* **46**, 449 (1967).

- [27] A. J. Menezes, P. C. van Oorshot, and S. A. Vanstone, *Handbook of Applied Cryptography* (CRC Press, Boca Raton, 1997).
- [28] N. Gershenfeld and G. Grinstein, *Phys. Rev. Lett.* **74**, 1970 (1995).
- [29] William L. Brogan, *Modern Control Theory* (Prentice Hall, Englewood Cliffs, NJ, 1991).
- [30] Mario di Bernardo, *Int. J. Bifurcation Chaos Appl. Sci. Eng.* **6**, 557 (1996).
- [31] C.-C. Chen, *Phys. Lett. A* **213**, 148 (1996).
- [32] G. Chen and X. Dong, *Int. J. Bifurcation Chaos Appl. Sci. Eng.* **3**, 1363 (1993).
- [33] T. L. Carroll, *Phys. Rev. E* **50**, 2580 (1994).
- [34] T. L. Carroll and L. M. Pecora, *IEEE Trans. Circuits Syst.* (to be published).
- [35] W. H. Press *et al.*, *Numerical Recipes* (Cambridge University Press, New York, 1990).
- [36] L. Kocarev and U. Parlitz, *Phys. Rev. Lett.* **74**, 5028 (1995).
- [37] U. Feldmann, M. Hasler, and W. Schwarz, *IEEE Int. Symp. Circuits Syst.* **2**, 680 (1995).
- [38] L. M. Pecora, *SPIE Proc.* **2038**, 2 (1993).
- [39] B. Schneier, *Applied Cryptography* (Wiley, New York, 1996).