Driving trajectories in complex systems

Elbert E. N. Macau* and Celso Grebogi
Institute for Plasma Research, University of Maryland, College Park, Maryland 20742
(Received 15 October 1998)

A new paradigm, which combines targeting type of control problem for chaotic systems with the techniques used in system control theory, is proposed. This paradigm is used to rapidly change the evolution of a complex system among desired behaviors. We point out how this paradigm can also be applied to nonlinear systems that do not present the characteristics of a complex system. [S1063-651X(99)06404-1]

PACS number(s): 05.45.−a

I. INTRODUCTION

A complex system is any system that presents involved behavior, and is hard to model by using the reductionist approach of successive subdivision, searching for ''elementary'' constituents [1]. Regarding the system's behavior, if it is a complex system, we might expect to find the following [2–5]: (i) a behavior that is neither completely ordered and predictable nor completely random and unpredictable; (ii) its evolution reveals patterns in which coherent structures develop at various scales, but do not exhibit elementary interconnections; (iii) the structures can show a hierarchical relationship, i.e., nontrivial structures over a wide range of scales can appear. Nature provides plenty of examples of these systems [1], in fields as diverse as biology, chemistry, geology, physics, and fluid mechanics. Some of the most frequently quoted examples of systems that exhibit apparent complex behavior are Rayleigh-Bernard convection [6], Belousov-Zhabotinsky reaction [7], Arecchi's optical experiment [8], neuron activity [9], and fluidized beds [10].

Usually, the features that are typical of a complex system appear in systems with many degrees of freedom [5]. This is the case for all the systems previously cited. What happens, in general, is that for these systems we have a situation where a large number of both attracting and unstable chaotic sets coexist. As a result, we can have a rich and varied dynamical behavior, where many competing behaviors can exist. When the system is evolving in the neighborhood of an attracting periodic set, it will exhibit an ''ordered'' behavior. This behavior changes to an apparently ''nonordered'' behavior when the system is evolving about the unstable sets. Thereby, the attractors themselves are responsible for the appearance of coherent structures, while the specific characteristics of each individual attractor, combined with its location relatively to the unstable sets, are responsible for the appearance of a hierarchy of structures.

Although the double rotor is a low-dimensional system, Ref. [5] showed that this system is a nice paradigm of a complex system, when it is subjected to random external noise. The complexity occurs because of both the large number of coexisting periodic attractors and the complicated basin structures of the double rotor over a wide range of parameter values. The unstable invariant chaotic sets are embedded in the fractal basin boundaries. With the exception of small open neighborhoods about the periodic attractors, the phase space is permeated by the fractal basin boundaries the dimensions of which are very close to the dimension of the phase space. Though the trajectory can spend arbitrarily long times in the neighborhood of one of the stable periodic behaviors, the external random noise applied to the system prevents the trajectories from settling permanently into any one of them. Thus, this system presents the same behavior as was previously described for a complex system [5].

Using the double rotor system as a paradigm of a complex system, Ref. [5] showed how to use small amplitude feedback control to influence and manipulate the behavior of the system such that its trajectories can be confined to the neighborhood of any desired attracting state (ordered behavior). In fact, the strategy envisaged by these authors (a) uses the natural evolution of the system that, because of the unstable chaotic sets in the boundaries, leads to trajectories that eventually approach the neighborhood of any one of the attracting sets, and then (b) employs a method of control to keep a trajectory around an attracting state despite the presence of noise. Thus, according to (a), starting from any initial condition, the system follows its natural evolution until it falls near the selected attractor about which we want the trajectory to be stabilized. When this happens, the procedure (b) is applied and the trajectory stays confined to the neighborhood of the desired attractor. This strategy works very well, but the average time to go from a generic initial condition to the desired attracting state can be forbiddingly long in practice. In the example presented by the authors, this time is typically about a thousand iterations (periods).

In this article, we introduce a control method that allows us to rapidly change the dynamical behavior of the complex system as wished. Using our method, the transport time needed to go from one ordered state to another is substantially reduced. Furthermore, with this method, we can both keep the system in any one of its dynamical regimes as long as one wishes, and switch rapidly among these regimes. More than a control method, we are proposing a new paradigm of system control. This paradigm combines the targeting type of control, used for chaotic systems [12], with the classical control system methods from the system control theory. We demonstrate that this paradigm results in powerful strategies that can be used in situations that so far are

*Also at the Laboratório de Integração e Testes, Instituto Nacional de Pesquisas Espaciais (INPE), Brazil. Electronic address: emacau@@glue.umd.edu or elbert@@lit.inpe.br.

difficult or even impossible to be handled by using just one of these methods alone.

To demonstrate our control strategy, we use as a paradigm of a low dimensional complex system the damped kicked single rotor with noise. The damped kicked single rotor was introduced by Zaslavskii [11], and it is also a very nice paradigm of a complex system if one adds noise, as was demonstrated by Feudel and Grebogi [13], besides the fact that it has a dimension that is smaller than the double rotor and, as a result, it is more easily treatable.

This paper is organized as follows. In the next section we review the single rotor with noise. In Sec. III, we introduce our control method, and present the results from upon using it. In Sec. IV, we discuss its advantages and give the conclusions.

II. SINGLE ROTOR WITH NOISE

The kicked singled rotor describes the time evolution of a mechanical pendulum that is being kicked at times nT, $n=1,2,\ldots,$ with a constant force f_0. From the differential equation of this mechanical system one can derive a map which is related to the state of the system just after each successive kick [14]:

$$x_{k+1}=x_k+y_k(\mathrm{mod}\,2\pi) \tag{1}$$

$$y_{k+1}=(1-\nu)y_k+f_0\sin(x_k+y_k),$$

where x corresponds to the phase and y to the angular velocity. f_0 is the force parameter, and ν is the damping parameter, measuring the energy dissipation of the system. The parameter ν varies between 0, for a Hamiltonian situation, with no damping, and 1, in the case of a very strong damping. The dynamics lies on the cylinder $[0,2\pi)X\mathbb{R}$. In the very strong damping ($\nu=1$) limit, the system reduces to a one-dimensional circle map with a zero rotation number, and it exhibits the Feigenbaum scenario to chaos [14]. The dynamics lies on the circle $[0,2\pi)$.

In the Hamiltonian case (no damping, $\nu=0$), we have the area-preserving standard map, which was studied by Chirikov [15] and by many other authors [16–19]. It has stable and unstable periodic orbits, Kolmogorov-Arnol'd-Moser (KAM) surfaces, and chaotic regions. Depending on the nonlinear parameter f_0, the regions of regular motion and the regions of chaotic motion are complexly interwoven. As the second equation of the map is also taken to be modulo 2π, the map of the cylinder reduces now to the map of the torus $[0,2\pi)\times[0,2\pi)$ to itself. As a consequence, each of the periodic orbits represents, in fact, a family of overlapping periodic orbits in which the velocity y differs by integer multiples of 2π. Because of the modulo 2π, all periodic orbits of a same family are located at the same location on the thorus.

If we now consider the Hamiltonian case but introduce a very small amount of dissipation (ν value close to zero), the symmetry in y is broken, and the motion again takes place on the cylinder $[0,2\pi)\times\mathbb{R}$. The periodic orbits become sinks and the chaotic Hamiltonian sets become unstable chaotic sets embedded in the basin boundaries separating the various sinks. The chaotic motion is hence replaced by long chaotic transients that occur before the trajectory is eventually

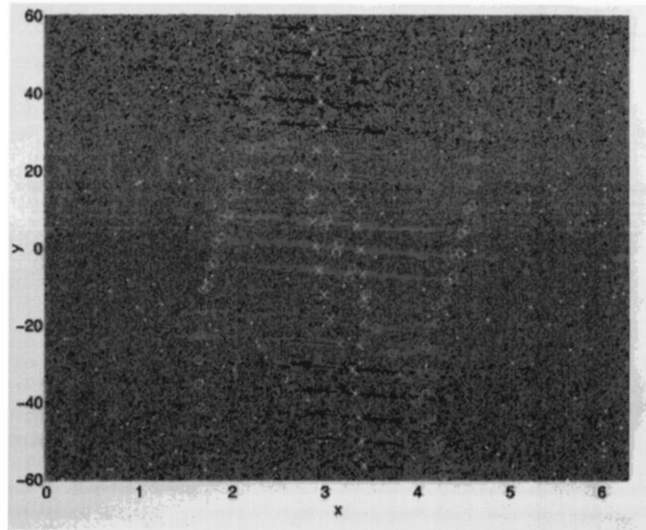

FIG. 1. Basin of attraction for the single rotor with noise. Asterisk indicates the position of attracting period-one points, × the position of attracting period-two orbits, and o the position of attracting period-three orbits. This picture is for the following parameters: $f_0=4.0$ and $\nu=0.02$. All quantities plotted are dimensionless.

asymptotic to one of the sinks [20]. Furthermore, the dissipation leads to a separation of the overlapping periodic orbits, that belong to a given family, with increasing modulo of the velocities on the cylinder. However, there is a bounded cylinder which contains all of the attractors [20]. This cylinder is given as $[0,2\pi)\times[-y_{max},y_{max}]$, where $y_{max}=f_0/\nu$, and all trajectories are eventually trapped inside this region [20]. Consequently, for values of ν close to zero, there is a large, but finite, number of coexisting periodic orbits of increasing period. Figure 1 is a picture in the space of initial conditions showing the basins of attraction for all attractors of period one to five. The periodicity of the attractors in the picture is distinguished by colors, while the locations of the attracting periodic orbits is identified by special characters that are mentioned in the figure caption.

Figure 2 shows a typical basin of attraction for a periodic attracting orbit. The black points are attracted to the specific attractor. The particular picture shows the basin of attraction for a fixed point at $y=6\pi$. The basins of attraction have fractal boundaries. Feudel *et al.* [20] calculated the *uncertainty exponent* (α), which measures the sensitivity of the final state to small changes in the initial conditions. This exponent is typically related to the box counting dimension d of the basin boundary by $\alpha=D-d$, where D is the dimension of the state space. For damping $\nu=0.05$, the result is $\alpha=0.00641$, which implies $d=1.99359$; for $\nu=0.02$, the result is $\alpha=0.001$, and $d=1.999$. This means that the dimension of the basin boundaries is nearly the dimension of the state space, and they are organized in a complexly interwoven structure, with chaotic saddles embedded in these basin boundaries [21]. Furthermore, extremely small changes in the initial conditions may shift a trajectory from one basin to another, which means that the system has high sensitivity to the final state. Thus, which attractor is eventually reached by a trajectory of the system depends strongly on the initial conditions. This phenomenon is called multistability [13]. In this scenario, typical trajectories, starting with arbitrary ini-

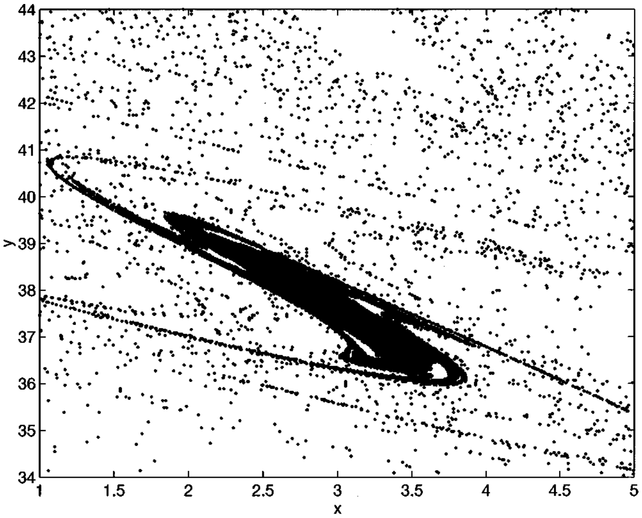

FIG. 2. Enlargement of the basin of attraction for a period one attracting orbit. Points located inside the region [1,5]×[34,44] that go to this period one attracting orbit are plotted. All quantities plotted are dimensionless.

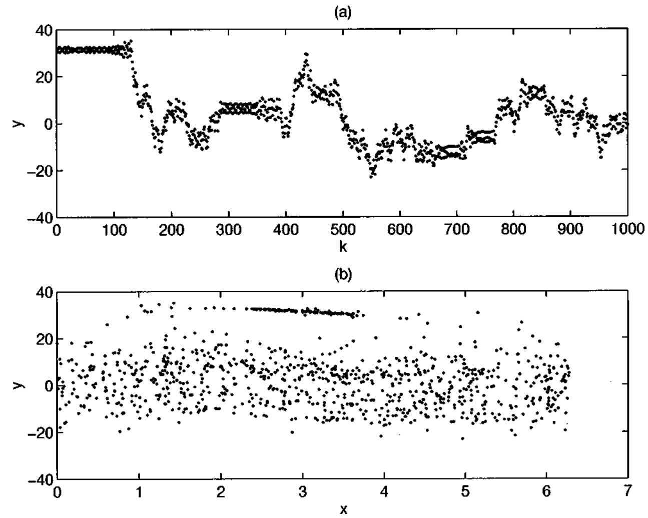

FIG. 3. (a) The y variable, which represents the angular velocity of the noisy kicked single rotor, vs the iteration number k for a typical noisy trajectory. (b) The same typical noisy trajectory plotted in the phase space. In both graphs, all quantities plotted are dimensionless.

tial conditions, experience periods of long chaotic transients before approaching one of the periodic attractors.

Consider now this previously described scenario, but in the presence of a small amplitude noise. Feudel *et al.* [20] showed that in this situation, called the *single rotor with noise*, the system can be characterized as a complex system [1], regardless of the fact that it is a system of low (just two!) dimensions. In fact, the noise may prevent the trajectories from settling into any of the stable periodic behaviors. The trajectory may come close to one of the periodic attractors, and remain in its neighborhood for some time. During this period, the trajectory's behavior is governed by the periodic attractor and it is, as a consequence, ordered. However, this ordered behavior just persists for a while, because noise will eventually move the trajectory out of this ''metastable'' state into the fractal boundary region. In the neighborhood of fractal basin boundaries, the trajectory's behavior is governed by the unstable invariant chaotic sets that are embedded there. As a consequence, the trajectory experiences a chaotic transient behavior for some time, until it approaches the same or another periodic attractor. The period of time that the trajectory is in the fractal boundaries corresponds to the trajectory's ''random'' behavior. Therefore, in a single rotor with noise, a typical trajectory alternates between intervals of random or chaotic motion and intervals of nearly periodic behavior. Figure 3 shows this behavior for a typical trajectory. Such behavior, that stresses the fact that the system is neither completely ordered and predictable nor completely random and unpredictable, has also been observed experimentally in Rayleigh-Benard convection [6], in coupled laser systems [8], and in fluidized beds [10]. In the figure, we also see that the trajectory visits the neighborhoods of different attractors in a ''random'' way. It is not possible to devise, for example, an empirical rule that allows one to forecast the sequence of attractors that will be visited by the noisy trajectory from the knowledge of the attractors previously visited. In the next section, we show how it is possible to drive trajectories of this complex system so that it reaches a desired behavior in the shortest possible time.

III. DRIVING TRAJECTORIES

Recently, Ref. [5] showed that for a complex system the unstable chaotic sets in the basin boundaries provide the necessary sensitivity and flexibility to drive the system dynamics toward a specific ''ordered'' behavior using small perturbations. By ''ordered'' behavior we mean the stabilization of one of the metastable attracting sets of the system [5]. Once a trajectory enters one of these open neighborhoods, it executes an almost periodic (''ordered'') motion until the noise takes the trajectory back to the basin boundary region. The method proposed by the authors in Ref. [5] leaves the system evolving by itself, until it comes close to the desired metastable state. When that happens, a judiciously chosen perturbation is applied so that the system is stabilized about the desired neighborhood of the metastable attracting set. The feedback perturbation is applied as soon as the system reaches a neighborhood of the desired metastable state. The method works as follows [5]: Let us consider the single rotor with noise described by

$$x_{n+1}=F(x_n)+\delta=\tilde{F}(x_n), \tag{2}$$

in which δ noise uniformly distributed is added to the system at each iteration. For simplicity, we assume that the metastable state to be stabilized is a fixed point x^*. We can linearize the system in the neighborhood of this point as

$$F(x^*+\epsilon)=x^*+DF(x^*)*\epsilon+\delta, \tag{3}$$

where $DF(x^*)$ is the derivative of the map F at x^*. The eigenvalues to $DF(x^*)$ are inside the unit circle, since x^* is stable without noise. Suppose that on the ith iterate, the trajectory lands in a neighborhood of this fixed point, so that $x_i=x^*+\epsilon$. Without control, the next point will be $x_{i+1}=F(x_i)+\delta$. Assuming that the linearization holds approximately about x^*, the fixed point can be stabilized by the addition of a controlling term $-DF(x^*)(x_i-x^*)$, so that the next point is now

$$\hat{x}_{i+1}=F(x_i)+\delta-DF(x^*)(x_i-x^*). \tag{4}$$

Since we want to achieve control using only small perturbations, the term $|DF(x^*)(x_i-x^*)|$ is scaled, when necessary, so that it does not exceed some predetermined upper bound previously specified. Note that the effect of this control law is to impose a superstable condition on the fixed point x^*.

This method works very well in stabilizing trajectories in the neighborhood of periodic attractors. Moreover, this method [5], which combines the ability of the system to reach a desired metastable state and the method to hold the system's evolution about that particular state, relies on the ability of a complex system to access many different states, and one's ability to modify the system's complex behavior by using only small perturbations.

The *transport time* involved in the process of changing the complex systems' evolution can be, typically, excessively long [5]. That happens because the method relies on the transport time until the system's evolution brings the trajectory close to the desired state before the stabilization strategy can be applied. Thus, for example, if the goal is to change the system's evolution from the metastable *A* to the metastable *B*, the orbit that accomplishes that experiences arbitrarily long chaotic transient in going from *A* to *B*. It may happen that, before reaching *B*, the trajectory might evolve about the periodic attractor *C*, until the moment that noise sends the trajectory back to the fractal basin boundary. There, the trajectory is again a chaotic transient. This kind of behavior may recur many times, until eventually the trajectory finally approaches the desired periodic attractor *B*, so that the stabilizing strategy of Ref. [5] can now be used to keep the trajectory evolving about the desired state. In practical situations, such transport time is prohibitively long.

We find and show in this paper that this transport time can be substantially reduced. As the transition between two different metastable states implies a trajectory that undergoes a chaotic transient evolution in between the states, we devise the idea of using a targeting method for the complex dynamics. The targeting method exploits the inherent exponential sensitivity of the chaotic time evolution to tiny perturbations and our ability to choose the right perturbations to direct trajectories to some desired accessible state in the shortest possible time [22]. To apply this method, suppose that we have two points *S* and *T* in the phase space of our complex system, both of them in the fractal basin boundary. Consider a small region r_S about the source point *S*, and another small region r_T about the target point *T*. The objective is to find a point p_S in r_S so that it belongs to a trajectory that goes from p_S to a point p_T in r_T. To find p_S, the region r_S is iterated in the forward direction, while the region r_T is iterated in the backward direction, until these iterated regions intersect each other at p_I, in the phase space. When the intersection is found, there is a trajectory that goes from region r_S to region r_T through the intersection p_I. The point p_S is then used to determine the value of the perturbation that must be applied to the system to bring it to p_S. As the system is in p_S, it evolves following its own dynamics to get p_T in the shortest possible time [22]. Then, another perturbation can be used, if necessary, to bring the system from p_T to the target point *T*.

We now argue that this targeting method, with two modifications, can be straightforwardly applied to complex systems. The first modification results from the way that our noisy system evolves and the procedure to find the intersection between the forward and backward dynamics of the regions r_s and r_t, respectively. Consider that we are at a given point x_i. The next point x_{i+1} is obtained through the following procedure. We iterate once the point x_i under the map, but for n different noise realizations, using for each noise realization the same x_i as the initial condition. Let us call $x_r(i+1,k)$ the point obtained when the map is iterated using the kth noise realization, so that

$$x_r(i+1,k)=\tilde{F}(x_i). \tag{5}$$

The result of this procedure is the sequence $x_r(i+1,1),x_r(i+1,2),\ldots,x_r(i+1,n)$. Then, x_{i+1} is calculated from this sequence of points by taking the average:

$$x_{i+1}=\frac{x_r(i,1)+x_r(i,2)+\cdots+x_r(i,n)}{n}. \tag{6}$$

This procedure deals with the additive noise in the single rotor. It is a filter algorithm to reduce the effect of this noise.

The other modification introduced related to the way the targeting procedure is applied in order for the phase space trajectory to go from the point p_S, near the source point *S*, to the point p_T, near the targeting point *T*. If we were dealing with a simpler chaotic system, starting from *S*, the only step that is necessary to get to the targeting point *T* would be the application of a small perturbation in *S* to move the state of the system to the point p_S. Then, the natural chaotic evolution of the system would guide the trajectory to the neighborhood of the targeting point p_T near *T*. For our complex system, the situation is different. If we applied the same procedure, starting from the point *S*, it is likely that the dynamics would not take the trajectory to the targeting point. That would happen because of the additive noise that exists in our system. The noise, in combination with the deterministic chaotic behavior of the map, implies a trajectory that is likely to deviate from the solely deterministic trajectory.

To deal with this more complicated behavior, we use the points from the deterministic trajectory obtained from the targeting method, but ''corrected'' according to the following procedure. Suppose that, starting from *S*, the deterministic trajectory is a sequence of m points in the phase space $(x_{s_1},x_{s_2},\ldots,x_{s_m})$, where x_{s_1} is the perturbed point p_S, and x_{s_m} is the point p_T. Starting from p_S, the ''correctional'' procedure consists of the application of successive perturbations to compensate for the natural deviation of the (noisy) system trajectory from the deterministic trajectory. Thus, for example, if at iteration *j*, the trajectory the system arrives at position x_j that is different from x_{s_j}, a perturbation is calculated and applied to the system to take the trajectory from x_j to the point x_{s_j}. Note that for our system this perturbation is of the same order as the noise added at each iteration of the map [see Eq. (1)].

In Figs. 4 and 5 we show the application of our procedure.

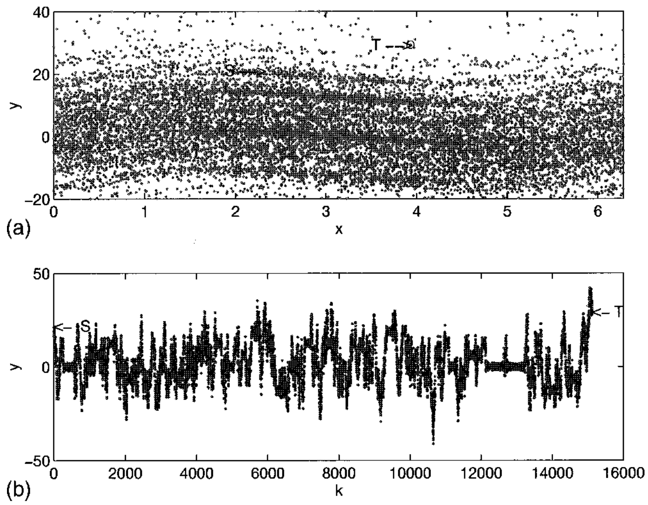

FIG. 4. (a) Phase space plot of a trajectory starting at the source point *S*, and after letting the system to evolve by itself; it eventually reaches the targeting point *T* after a large number of iterations. (b) This picture shows the *y* component of a trajectory in phase space going from the source point *S* to the target point *T*. It depicts then the angular velocity *y* of the noisy kicked single rotor as a function of the iteration number *k*. In both graphs, all quantities plotted are dimensionless.

Both the source point *S* and the targeting point *T* are in the fractal basin boundary. In Fig. 4, starting from the source point, we leave the system evolving by itself, until it comes close to the targeting point. After 15 118 iterations, the trajectory visits a small region about the target *T*. This transport time can be substantially reduced by applying our targeting procedure. Our method, for this case, permits the target to be attained in 30 iterations, as can be seen in Fig. 5. Thus, we achieve an improvement of three orders of magnitude in reducing the transport time, which is a very significant result. It must be stressed that this result is a consequence of the ability of the complex system to reach many different states, combined with its sensitivity to small perturbations due to the unstable chaotic set embedded in the random structure.

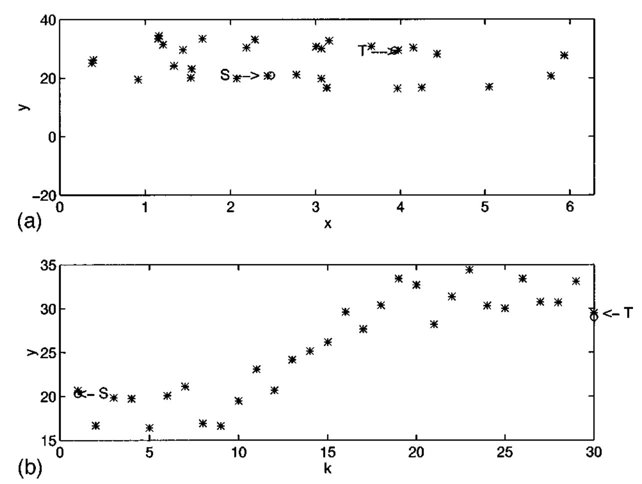

FIG. 5. By exploring the chaotic behavior of the system, our targeting procedure rapidly steers the trajectory from *S* to *T*. The asterisks that appear in the picture represent a trajectory obtained by applying our targeting procedure to drive the system from the point *S* to the point *T*. (a) Phase space plot. (b) The *y* component of this trajectory as a function of *k*. In both graphs, all quantities plotted are dimensionless.

The procedure just described works for points located in the random structure. The targeting methods, ours and the previous one [22], work because of the inherent exponential sensitivity of the chaotic time evolution to perturbations. Therefore, the source point *S* and the targeting point *T must* both be in the same chaotic invariant set. This is the case for points *S* and *T* in Figs. 4 and 5. However, if the system is evolving in a metastable regime, where the trajectory can be trapped for an arbitrarily long time, the condition of being located in the same random structure is not satisfied. Furthermore, the time evolution is ''ordered,'' and the inherent exponential sensitivity to perturbations does not apply. Thus, we show next that if the objective is to bring the trajectory from one metastable state to any other metastable state, first the trajectory must be guided to the random chaotic region where the targeting method is applied. The idea then is (i) to remove the trajectory from the metastable region to the surrounding random structure, (ii) apply the targeting procedure in the random structure to bring it to the neighborhood of the desired metastable state and then finally (iii) to bring it to the desired targeting point inside the metastable state. We accomplish this guidance task inside a metastable state [(i) and (iii)] by using a traditional technique from the system control theory and outside the metastable states (in the chaotic invariant region) (ii) using the targeting procedure just described. Thereby, our method of targeting different states in a complex system is a combination of a modified (to account for the effect of noise) chaotic targeting with the traditional control method.

For the traditional part of the targeting procedure, we use the *discrete linear quadratic regulator* (DLQR) [23]. We consider, for simplicity, the system evolving in a metastable state about, say, a fixed point x^*. Linearizing the system about this point gives the following equation:

$$x_{k+1}=Ax_k, \tag{7}$$

where A is $DF(x^*)$. To change the state of the system, we introduce an input term into Eq. (7) as

$$x_{k+1}=Ax_k+Bu_k, \tag{8}$$

where u_k is the vector of inputs and B is a constant matrix that states how the inputs influence the state of the system. Our objective is to pick u_k so that the ''cost function''

$$J=\tfrac{1}{2}\sum_{k=0,N}(x_k^t Q_1 x_k+u_k^t Q_2 u_k) \tag{9}$$

is minimized. $Q1$ and $Q2$ are symmetric weighting matrices to be selected, as we show next, based on the relative importance of the various states and controls. A particular weight is almost always selected for the control ($|Q_2|\neq 0$), to avoid large components in the control gains. The Q's must also be non-negative.

We solve this control problem by minimizing Eq. (9) subjected to the constraint Eq. (8),

$$-x_{k+1}+Ax_k+Bu_k=0,\quad k=0,1,\ldots,N. \tag{10}$$

We use the method of Lagrange multipliers [24] to solve the problem by introducing one Lagrange multiplier vector λ_{k+1} for each value of *k*. The minimization leads to the following equations:

$$u_k^t Q_2 + \lambda_{k+1}^t B = 0, \tag{11}$$

$$-x_{k+1} + A x_k + B u_k = 0, \tag{12}$$

$$\lambda_k = A^t \lambda_{k+1} + Q_1 x_k . \tag{13}$$

Combining Eqs. (10), (11), and (13) gives us a set of coupled difference equations defining the optimal solution of x_k and λ_k and hence u_k, provided the initial (or final) conditions are known. The initial conditions x_0 must be given; however, usually λ_0 is not known, and we need the endpoint x_n to establish the final condition. From Eq. (9), we see that u_N is zero for the minimum *J* since u_N has no effect on x_N [see Eq. (10)]. Thus, Eq. (11) suggests that $\lambda_{N+1}=0$, and Eq. (13) thus shows that a suitable condition is

$$\lambda_N = Q_1 x_N . \tag{14}$$

The solution to the optimal control problem is now completely specified. It consists of the two difference equations, Eqs. (10) and (13), with u_k given by Eq. (11), the final condition on λ given by Eq. (14), and the initial condition x_0 assumed to be given in the statement of the problem. Problems like this, where a set of ordinary differential equations or difference equations is required to satisfy boundary conditions at more than one value of the independent variable, are called *two point boundary value problems*. The solution to this two point boundary-value problem is not so easy. One way to solve this problem is proposed by Bryson and Ho [25], in which it is assumed that

$$\lambda_k = S_k x_k , \tag{15}$$

where S_k is an arbitrary matrix. Introducing this assumption into Eq. (11), and after some work [25], we get

$$S_k = A^t M_{k+1} A + Q_1 , \tag{16}$$

and

$$u_k = -K_k x_k , \tag{17}$$

where

$$M_{k+1} = S_{k+1} - S_{k+1} B (Q_2 + B^t S_{k+1} B)^{-1} B^t S_{k+1} , \tag{18}$$

and

$$K_k = (Q_2 + B^t S_{k+1} B)^{-1} B^t S_{k+1} A. \tag{19}$$

In these equations, the boundary condition on the recursive relationship for S_k is obtained from Eqs. (14) and (15); thus

$$S_N = Q_1 . \tag{20}$$

K_k is the desired time-varying feedback gain. Note that the gain K_k changes for each step but can be precomputed and stored for later use as long as the length *N* of the problem is known. Note that no knowledge of the initial state x_0 is required for computation of the control gain K_k, and the Eq. (16), which gives S_k, must be solved backward, with the initial condition that appears in Eq. (20), plus the condition $K_N=0$.

This closed loop control strategy, when applied to a linear system, has the effect of bringing this system from an arbitrary initial state to the zero state as quickly as possible. After that, the system is held in the zero state, even in the presence of noise. If the desired final state is instead x_f, we shift the origin. Let u_f be the constant input signal to which x_f corresponds to the steady-state output. Then, u_f and x_f are related by

$$x_f = A x_f + B u_f . \tag{21}$$

We introduce now the state space variable

$$\hat{x}_k = x_k - x_f . \tag{22}$$

Then, with the aid of Eq. (21), it follows from Eq. (8) that $\hat{x}_k$ satisfies the equation

$$\hat{x}_{k+1} = A \hat{x}_k + B \hat{u}_k , \tag{23}$$

where

$$\hat{u}_k = u_k - u_f . \tag{24}$$

This shows that the problem of bringing the system (8) from an arbitrary initial state x_0 to the final state x_f is equivalent to bringing the system (8) from the initial state $x_0 - x_f$ to the equilibrium state $x_k=0$.

In Figs. 6 and 7 we show the result of using the DLQR method on the complex system. The system is initialy evolving in a metastable state about a fixed point that exists when there is no additive noise. This situation is represented in the phase space plot, Fig. 6, by dots. Our initial goal is to steer the trajectory, say from *S* to a neighborhood of the targeting point *T*, as indicated in Fig. 6. Both points *S* and *T* belongs to the same metastable region, but *T* is located near the surrounding random structure. From *T*, the trajectory can be guided to evolve in the random structure by using a small perturbation. Using the position of *T* as reference, and linearizing the system in the neighborhood of the fixed point, we calculate the feedback gain that should be used so that the DLQR can stabilize the trajectory evolution about *T*. When the trajectory visits the point that is indicated in the figures by *S*, the DLQR controller is activated, using the previously calculated feedback gain. The effect of applying this controller can be seen in Fig. 6, where the controlled trajectory is represented by circles. The controller changes the system's dynamics so that, after a transient, it starts to evolve about *T*. In Fig. 7, we represent the distance of the trajectory from the targeting as a function of *k*, starting from the instant when the DLQR is applied. We can see in this figure that the controller brings the trajectory very near the targeting point *T*, as desired. However, after evolving for some time about *T*, the trajectory moves progressively away from it. This happens because of the combined effects due to nonlinearity and noise. This progressive deviation from the desired state is more intense as the distance of the desired final state from the point that was used for the linearization increases.

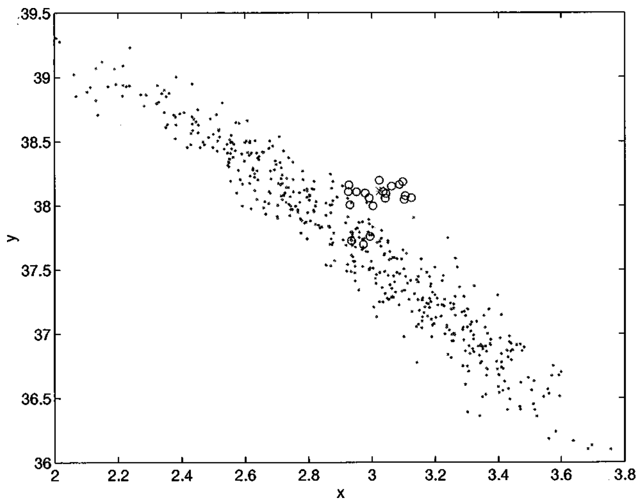

FIG. 6. The effect of using the DLQR. The system was evolving in a metastable state, represented by dots in the phase space. When it is in the position *S*, represented by a + in the graph, the DLQR was applied. The effect of using this controller is shown with circles in the figure. The control changes the system dynamics so that the system passes to evolve about the targeting point *T*, represented in the graph by an asterisk. All quantities plotted are dimensionless.

The behavior of the system under the action of the DLQR, however, is precisely what we need to take the system evolution out of a metastable state into the random structure in order to drive the system's trajectory toward another specific metastable state. Thus, suppose the system is in a metastable state and we wish to stir its dynamics out of this state to a random structure. We first identify in the state space the point x_f inside the metastable region, the region where the system is currently evolving, but an x_f very near the random structure. Then, we apply the DLQR, considering x_f as the desired final state. When the trajectory is in the point x_{f_n}, which is located close enough to x_f, we cease to use the DLQR and immediately apply a small perturbation to drive the system to x_f. Then, another perturbation takes the system from x_f to the neighboring random structure as desired. In fact, it is sufficient to give a single perturbation to take the system from x_{f_n} to the random structure and, thus, bypassing the point x_f altogether.

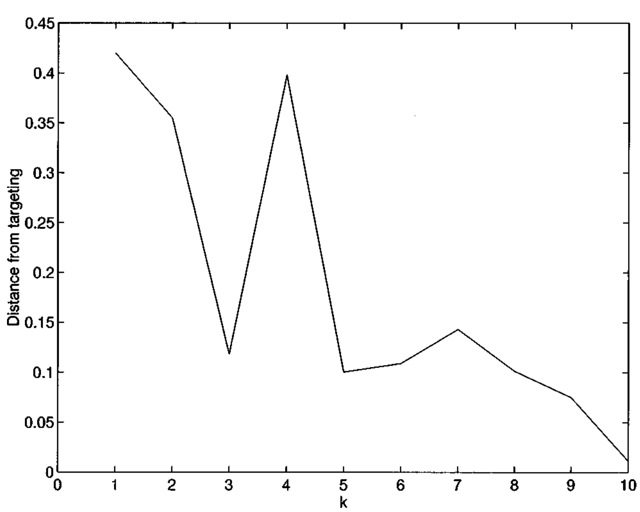

FIG. 7. This plot shows the distance of the DLQR controlled trajectory from the target as function of *k*. When the controlled trajectory comes near enough to the target, the application of DLQR can be stopped and a small perturbation is enough to send the trajectory to the random structure. All quantities plotted are dimensionless.

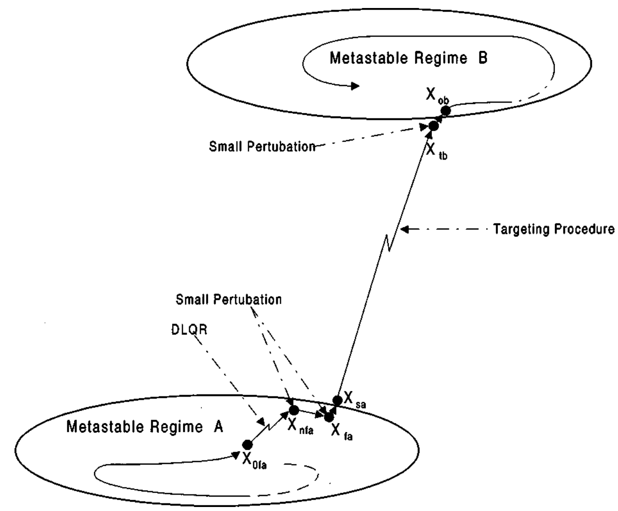

FIG. 8. Schematic representation of our complete targeting procedure for complex systems. The system is initially evolving in the metastable regime *A*. Our goal is to steer it to metastable regime *B*. Starting from a generic point x_{0fa} in *A*, the DLQR procedure drives the trajectory inside *A* to a point x_{nfa} near x_{fa}. In x_{nfa} a small perturbation is applied, and the system moves to the state x_{fa}. Another perturbation is applied, and the system moves to the state x_{sa}. Our modified chaotic targeting procedure is then used to stir the system to x_{tb}. Another small perturbation drives the system to the point x_{ob}, where the procedure of Ref. [5] stabilizes the system in the metastable regime *B*.

Putting all these ideas together, imagine now that our goal is to stir the system from a state in the metastable regime *A* to a state in the metastable regime *B*, as shown in Fig. 8. Using the combined targeting procedure, we first calculate a trajectory in the state space that takes the point x_{sa} near the metastable regime *A* to a point x_{tb} near the metastable regime *B*. x_{sa} is chosen so that there is a point x_{fa} inside *A* so that a small perturbation is enough to drive the system from x_{fa} to x_{sa}. Also, x_{tb} is chosen in the random structure so that a small perturbation is enough to drive the system from x_{tb} to a point x_{ob} in *B*. Starting from a point in *A*, say, the point x_{0fa}, the DLQR procedure drives the trajectory inside *A* to a point x_{nfa} near x_{fa}. At x_{nfa} a small perturbation is applied, and the system instantaneously moves to the state x_{fa}. Another perturbation is applied, and the system instantaneously moves to the state x_{sa}. Our modified chaotic targeting procedure is then used to stir the system to x_{tb}. Another small perturbation instantaneously drives the system to the point x_{ob}, where the procedure of Ref. [5] stabilizes the system in the metastable regime *B*.

Relevant questions about our procedure are (1) how should the points x_{fa} and x_{sa}, as shown in Fig. 8, be chosen; and (2) what is the small perturbation that can move the system from one point to the other? This selection is made by using a procedure [26] used to find an acessible point [24]. The metastable state evolves about a periodic attractor

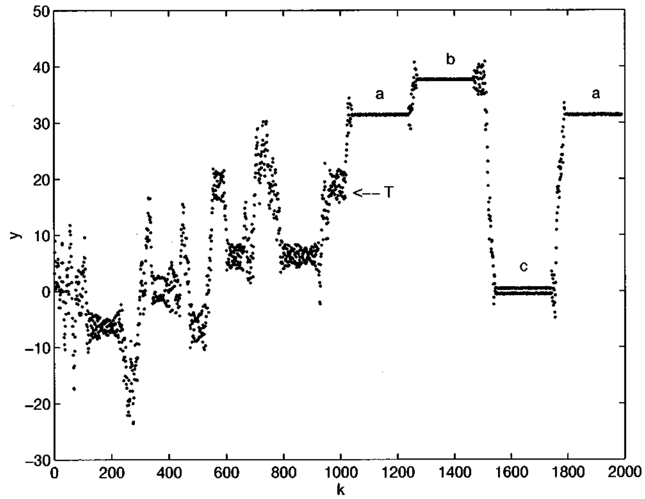

FIG. 9. Controlling the evolution of the complex system using a combination of our modified chaotic targeting method with standard targeting from control theory. The angular velocity of the noisy kicked single rotor as a function of k is plotted. The point T indicates the position where we start to control the system. Using our procedure, we drive the system through the metastable states that are indicated in the graph. The metastable regimes are identifying by small letters, while the point where the targeting procedure is applied for the first time is indicated by the letter T. All quantities plotted are dimensionless.

for the deterministic dynamics. The basin boundary of this periodic attractor permeates most of the state space, except for a small open neighborhood about the periodic attractor, as can be seen in Fig. 2. Then, points in this open neighborhood remains close to the periodic attractor, while points that are located outside undergo a chaotic transient, and eventually go to the neighborhood of other periodic attractors. To find x_{fa} and x_{sa}, we first find a point x_{ma} between both that is on the boundary but acessible from the basin of the periodic attractor. A point on the boundary is accessible if it is possible to connect this point to the attractor using a curve of finite length. Consider two points P_i, and P_o, one located inside this open neighborhood region about the periodic attractor, and the other outside, respectively. The point P_i approaches the attractor asymptotically while the point P_o does not. Take the middle point between P_i and P_o and call it P_m. Check then whether P_m is an inside or outside point. If P_m is inside, we discard P_i and take P_m as the new P_i. Otherwise, we discard P_o and take P_m as the new P_o. By repeating this procedure we can get two points P_i and P_o that are as near to each other as we wish, thus zooming in on an accessible point. We then rename the resulting P_i as x_{fa}, and P_o as x_{sa}. In the presence of the additive noise, the same procedure can be used, with Eq. (6) being considered instead.

In Figs. 9 and 10 we show the results of applying our combined method to change the system evolution at will between desired metastable states. We follow a typical noisy trajectory for more than 1000 iterations. Then, we apply our combined targeting procedure to drive the trajectory to evolve about a periodic attractor. The Ref. [5] method is then used and the trajectory is stabilized in this metastable regime. After some iterations, the DLQR is applied, followed

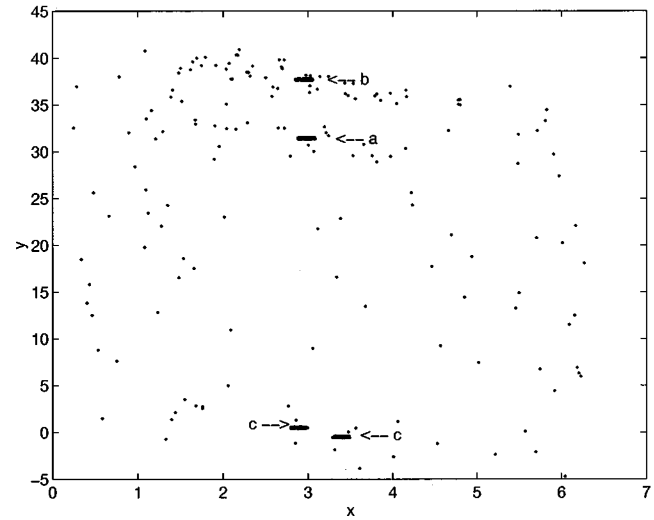

FIG. 10. Phase space plot of the evolution of the complex system using a combination of our modified chaotic targeting method with standard targeting from control theory. The metastable states where the system is stabilized are indicated in the plot. All quantities plotted are dimensionless.

by a targeting procedure, and the trajectory is stirred to the neighborhood of another metastable state. The same procedure is again applied to move the system now to a period two metastable state, where it is stabilized. After while, the procedure is applied to return the system to a previous metastable regime. The perturbations that is applied during the whole manipulation is less than 0.1. We note the extremely short transient in between the various controlled metastable states.

IV. CONCLUSION

The control method discussed in this article combines techniques used in system control theory with the targeting type of control for chaotic systems. The result of this combination is a method that can be used to rapidly change the time evolution of a complex system as desired. However, more than just a control method, what we are proposing is a new paradigm for manipulating and controlling the systems' dynamics. It is a combination of chaos control, control system strategies using small perturbation, and the standard control approach.

The efficacy of this proposed paradigm is demonstrated by using it in a complex system. A complex system, with its complicated and intricate dynamics, intrinsic sensitivity, and coexistence of states with different behavior, provides an ideal scenario to be explored by our paradigm. However, we expect that it can be used with the same efficacy in the control of other nonlinear systems where complicated dynamics occur. For these systems, a mechanism that switches the dynamics between chaos and regular motion can be used to provide the perfect scenario for the use of our paradigm.

This work was supported by the CNPq (Brazilian Agency for Research and Technological Development) and NSF/CNPq joint research grant. E. M. thanks Drs. Leon Poon and Ricardo Viana for helpful discussions.

[1] R. Badii, and A. Politi, *Complexity: Hierarchical Structures and Scaling in Physics* (Cambridge University Press, Cambridge, 1997).
[2] P. Grassberger, in *Fifth Mexican School on Statistical Mechanics*, edited by F. Ramos-Gómez (World Scientific, Singapore, 1991), p. 57.
[3] J. P. Crutchfield and K. Young, Phys. Rev. Lett. **63**, 105 (1989).
[4] J. Horgan, Sci. Am. **272**(6), 104 (1995).
[5] L. Poon and C. Grebogi, Phys. Rev. Lett. **75**, 4023 (1995).
[6] P. Bergé and M. Dubois, Phys. Rev. Lett. **93A**, 365 (1983).
[7] S. C. Müller and B. Hess, in *Cell to Cell Signalling: from Experiments to Theoretical Models*, edited by A. Goldbeter (Academic, London, 1989), p. 503.
[8] F. T. Arechi, G. Giacomelli, P. L. Ramazza, and S. Residori, Phys. Rev. Lett. **65**, 2531 (1990).
[9] P. E. Rapp, I. D. Zimmerman, E. P. Vining, and N. Cohen, J. Neurosci. **14**, 4731 (1994).
[10] C. S. Daw, C. E. A. Finney, M. Vasudevan, N. A. van Goor, K. Nguyen, D. D. Bruns, E. J. Kostelich, C. Grebogi, E. Ott, and J. A. Yorke, Phys. Rev. Lett. **75**, 2308 (1995).
[11] G. M. Zaslavskii, Phys. Lett. **A69**, 145 (1978).
[12] C. Grebogi and Y. C. Lai, Control Lett. **31**, 307 (1997).
[13] U. Feudel and C. Grebogi, Chaos **7**, 597 (1997).
[14] G. Schmidt and B. W. Wang, Phys. Rev. A **32**, 2994 (1985).
[15] B. V. Chirikov, Phys. Rep. **52**, 265 (1979).
[16] J. M. Greene, J. Math. Phys. **20**, 1183 (1979).
[17] G. Schmidt, Phys. Rev. A **A22**, 2849 (1980).
[18] A. J. Lichtemberg and M. A. Lieberman, *Regular and Chaotic Dynamics* (Springer-Verlag, New York, 1992).
[19] J. D. Meiss, J. R. Lary, J. D. Crawford, C. Grebogi, A. N. Kaufway, and H. D. I. Abrabanel, Physica D **6D**, 375 (1983).
[20] U. Feudel, C. Grebogi, B. Hunt, and J. A. Yorke, Phys. Rev. E **54**, 71 (1996).
[21] C. Grebogi, H. Nusse, E. Ott, and J. A. Yorke, in *Dynamical Systems* **1342**, edited by J. C. Alexander (Springer-Verlag, New York, 1988), pp. 220–250.
[22] T. Shinbrot, E. Ott, C. Grebogi, and J. A. Yorke, Phys. Rev. Lett. **65**, 3215 (1990).
[23] K. J. Aström and B. Wittenmark, *Computer Controlled System: Theory and Design* (Prentice-Hall, Englewood Cliffs, New Jersey, 1984).
[24] C. Grebogi, E. Ott, and J. A. York, Physica D **24**, 243 (1987).
[25] A. E. Bryson, and Y. C. Ho, *Applied Optimal Control* (Blaisdell, Waltham, Massachusetts, 1969).
[26] P. N. Battelino, C. Grebogi, E. Ott, J. A. Yorke, and E. D. Yorke, Physica D **32**, 296 (1988).