Forcing the escape: Partial control of escaping orbits from a transient chaotic region

Gaspar Alfaro · Rubén Capeáns · Miguel A.F. Sanjuán

Abstract A new control algorithm based on the partial control method has been developed. The general situation we are considering is an orbit starting in a certain phase space region Q having a chaotic transient behavior affected by noise, so that the orbit will definitely escape from Q in an unpredictable number of iterations. Thus, the goal of the algorithm is to control in a predictable manner when to escape. While partial control has been used as a way to avoid escapes, here we want to adapt it to force the escape in a controlled manner. We have introduced new tools such as escape functions and escape sets that once computed makes the control of the orbit straightforward. We have applied the new idea to three different cases in order to illustrate the various application possibilities of this new algorithm.

Keywords Controlling chaos, partial control, transient chaos, escaping orbits

1 Introduction

Even though chaotic systems are difficult to deal with due to its intrinsic unpredictability, there are nonetheless methods that allow us to control them. Different techniques for controlling chaos have been developed in the past few years. A rough classification may divide them between feedback control and non-feedback control methods. Among the first, we can consider the OGY [1] or the Pyragas [2] control methods, while on the latter random, chaotic or periodical signals are used as an appropriate mechanism to control the system.

The partial control method, which is a feedback method, has been used in previous works [3–9], and is applied to a map defined in a certain region Q where there is transient chaos and in absence of any control the orbit will eventually escape from the region after a certain number of iterations. Furthermore, the map is subjected to a disturbance which is always larger than the applied control. The goal is to use the minimum control to keep the orbits inside Q in presence of the disturbance. Precisely, one advantage of this method is the capacity to keep small the amount of control. As is well known, transient chaos is the physical manifestation of the the presence in phase space of a chaotic saddle, which is a fractal set. Orbits starting close to the chaotic saddle eventually escape in a highly unpredictable manner, and the escape times also depend on the disturbance and the initial conditions. In any case, one key feature of the method is that the control used is always smaller than the disturbance, which is rather surprising and counterintuitive.

In the present work, we face a new objective, which can be viewed as the converse of the previous one. While in the previous case the goal was to keep the orbit in Q for ever, now based on the same premises, our goal is to control the number of iterations necessary for the orbit to escape Q.

We have analyzed three different scenarios to apply the new strategy. The first one corresponds to the case where the controller wants to force the escape of the orbits from Q in N or less iterations of the map. In the second case, we consider the situation when we want the orbit to escape in exactly N iterations, where this stronger constraint would necessarily imply a higher amount of control. Finally, the third situation we con-

G. Alfaro · R. Capeáns · M.A.F. Sanjuán
Nonlinear Dynamics, Chaos and Complex Systems Group, Departamento de Física, Universidad Rey Juan Carlos, Móstoles, Madrid, Tulipán s/n, 28933, Spain
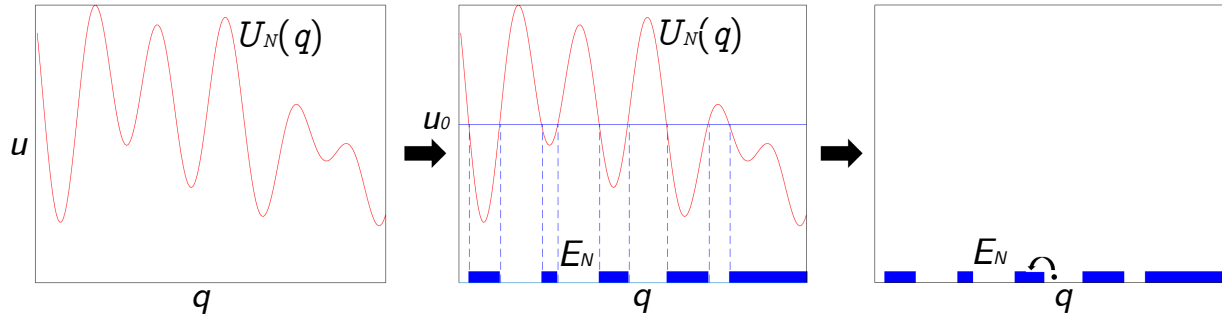E-mail: miguel.sanjuan@urjc.es

Fig. 1 Steps that summarize the partial control procedure. (Left) Escape function $U_N(q)$. For a given initial point $q \in Q$, this function represents the minimum upper control bound u_0 necessary to expel an orbit out of the region Q in N iterations. (Center) Escape set E_N (blue boxes) obtained as a result of the intersection from the escape function with the control bound u_0 (horizontal blue line). The points in E_N will escape from Q by using controls $|u_n| \leq u_0$. (Right) A point outside the escape set is controlled to insert it in the set. By doing so, each iteration guarantees that the orbit will escape in N iterations or less.

sider is somehow different. We have a map defined in two different regions in phase space, so that for a value of the parameter two chaotic attractors coexist independently in each region, and after a certain parameter value both attractors merge into a single global chaotic attractor. The idea here is to apply our control technique to fix the precise number of iterations of the orbit to stay in each region. As a result, we will get a chaotic orbit that periodically oscillates between the regions. Obviously this scheme could be generalized to a larger number of regions.

The paper is organized as follows. In Sect. 2, we introduce the partial control method and explain how to use it to control the escape from the chaotic region. Furthermore, we describe the escape functions and escape sets, adapted from previous work [10] that have been used for our objectives. In Sect. 3, we present two different specific cases where we apply the algorithm to fix the number of iterations for the orbits to escape. In Sect. 4, we address the case where the algorithm is applied for the goal of alternating the orbit between two regions in a predictable manner. Finally, the main conclusions are provided in the last section.

2 The partial control method

We present here the partial control method [6,9] that is applied on maps in the following manner

$$q_{n+1} = f(q_n) + \xi_n + u_n$$
$$|\xi_n| \leq \xi_0 \tag{1}$$
$$|u_n| \leq u_0 < \xi_0,$$

where the map acts on values of $q \in Q$, Q is a region in phase space, ξ_n represents a bounded disturbance affecting the map at each iteration and u_n is the applied

control at each iteration, which is also bounded and importantly, smaller than the disturbance.

Our goal here is to perturb the orbit starting in Q by applying a sequence of controls $(u_1, u_2, .., u_N)$ in order to push the orbit out of Q in N iterations. Needless to say, we can achieve this objective by using different sequences, and the approach of partial control is to find the strategy that minimizes the upper bound of that sequence, that is, the $\min \big(\max \big(|u_1|, |u_2|, .., |u_N| \big) \big) = U_N(q)$, where q is the initial point of the orbit and U_N is the *escape function*. Once the escape function is computed, we can choose an upper control bound value u_0 and select the set of points $q \in Q$ that satisfy $U_N(q) \leq u_0$. We name this set, *the escape set E_N*. Any orbit starting in this set can be expelled from Q by using a sequence of N controls u_n $(n \leq N)$ with magnitude equal or smaller than u_0. The notions of escape functions and escape sets have been adapted from the safe functions and safe sets defined in [10]. The steps to apply this control technique are summarized as follows

1. Choose the phase space region Q where the control method will be applied. We assume that we know the map and the upper disturbance bound ξ_0 affecting it.
2. Compute the escape functions U_k with $k = 1 : N$ in the region Q. Remind that N is the number of iterations we need to expel the orbit out of Q.
3. Set the value u_0 and for every escape function U_k, compute the corresponding escape set E_k.
4. For every iteration of the map, we choose the appropriate u_n, $|u_n| \leq u_0$, to bring the orbit to the escape set E_k. Thus, we use the control so that the first iteration of the map brings the orbit to the escape set E_N, the second iteration of the map to the escape set E_{N-1} and so on, until the orbit escapes out of the region Q.

The steps 2, 3 and 4 of this procedure are illustrated in Fig. 1. The second step is the most computationally expensive, where the escape functions U_k are computed by using an algorithm developed in [10]. The algorithm is based on the observation that this kind of control problems can be solved backwards, starting from the last iteration. We will see that it is straightforward to compute the first function U_1 and through an iterative procedure obtain the rest of the escape functions U_k, since $U_{k+1} = f(U_k)$.
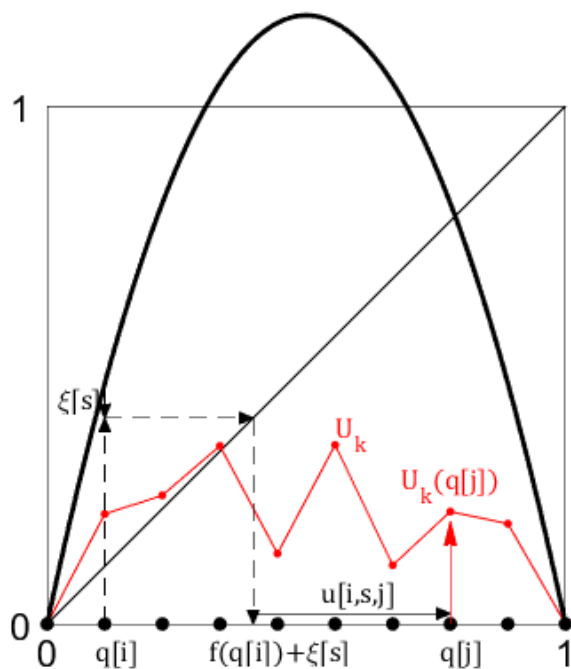

Fig. 2 Scheme of the escape function (red) and the terminology used in the control procedure. Here the logistic map with $\mu = 4.7$ is plotted. It is clear that orbits starting in points of $[0, 1]$ will escape the interval $[0, 1]$ after a few iterations. The dashed arrows are the mapping of an initial point $q[i]$ for one of the possible values of the perturbation $\xi[s]$. The horizontal black arrow represents the applied control that corresponds to the distance from the mapped point $f(q[i]) + \xi[s]$ to the arrival point $q[j] = f(q[i]) + \xi[s] + u[i, s, j]$. The indices i, j refer to points in the discretization of the interval $[0, 1]$, while s refers to values of the discretization of ξ_n.

Since we are doing numerical simulations, we must use a grid on Q so that the map becomes

$$q_{n+1} = f(q_n) + \xi_n + u_n \quad \rightarrow \quad q[j] = f(q[i]) + \xi[s] + u[i, s, j], \tag{2}$$

where $i = 1 : M$ denotes the number of the grid points in Q. The index $s = 1 : W$ corresponds with the number of possible disturbances ranging from $-\xi_0$

to ξ_0. The index $j = 1 : M$ denotes the arrival point $q[j] = f(q[i]) + \xi[s] + u[i, s, j]$. The term $u[i, s, j]$ denotes the control applied to the point $f(q[i]), \xi[s])$ to put it in the arrival point $q[j]$. All these terms are illustrated in Fig. 2 for clarity.

3 Escaping from the chaotic region

We will use the well-known logistic map $x_{n+1} = f(x_n) = \mu x_n(1 - x_n)$ as an example to illustrate the application of the algorithm described earlier.

When we consider values of $\mu > 4$, the logistic map presents transient chaos in the region $Q = [0, 1]$. On Fig. 3(a), the logistic map and one escaping orbit are represented. The number of iterations for which an orbit stays in Q without control depends on the initial condition and the sequence of disturbances affecting it. As a consequence the lifetime is highly unpredictable as shown in Fig. 3(b).

The goal of the algorithm based on the partial control method used in this work is to fix the number N of the iterations after which the orbit escapes from Q. Three different cases will be explored. In each case, we will show the algorithm to compute the corresponding escape functions with some examples.

3.1 Case A: Escape in N or less iterations

This situation is justified in the case when we want an orbit to leave the chaotic region as quickly as possible. To do that, we choose the value N and design the control algorithm so that the controlled orbit will abandon Q in N or less iterations. The lesser the value of N, the quicker the orbits will escape, though at a high price of control.

As already commented, the escape functions U_k are calculated from the first escape function U_1 through a recursive algorithm, since $U_{k+1} = f(U_k)$. Then, the value $U_1(q[i])$ is defined to be the minimum control necessary to escape in the next iteration. This control corresponds to the distance from the mapped point $f(q[i]) + \xi[s]$ to the nearest end points of Q (In this case 0 or 1). However, since there is a control $u[i, s, j]$ associated to each different value of the disturbance $\xi[s]$, we need to choose the maximum control among them all. Proceeding similarly for every initial condition $q[i]$, we obtain the escape function U_1. This function represents the minimum control bound necessary to escape in one iteration.

The next function U_2 will correspond to the minimum control bound necessary to force the escape of
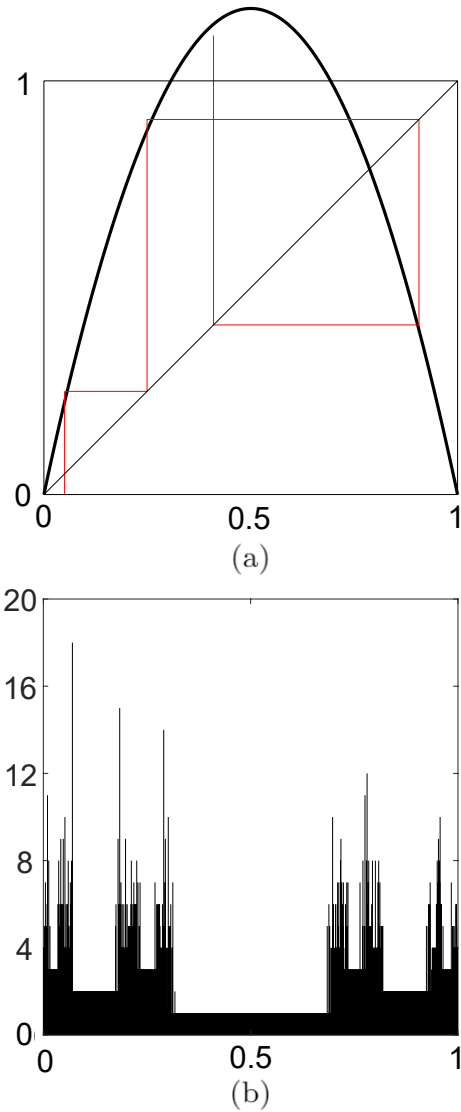
(a)

(b)

Fig. 3 (a) Logistic map with $\mu = 4.7$ and one orbit that eventually escapes from the region $Q = [0, 1]$. (b) Escape time for the noise-affected logistic map. It clearly presents a fractal structure, so the time an orbit will escape is unpredictable.

the orbit within 2 iterations. There are two possibilities here, given an initial condition $q[i]$, the image $f(q[i]) + \xi[s]$ can fall directly outside Q, in which case no control is needed, or can fall in a point $q[j] \in Q$. In the latter case, we need to compute the suitable control that minimizes the maximum between the values $u[i, s, j]$ (the control applied in this iteration of the map) and $U_1(q[j])$ (the maximum control that we will apply in the next iteration of the map), that is, the $\min_j (\max(u[i, s, j], U_1[j]))$. Again, to take into account all possible disturbances $\xi[s]$ we must choose

the maximum among all corresponding controls, that is $U_2(q[i]) = \max_s(\min_j(\max(u[i, s, j], U_1(q[j]))))$.

The procedure to compute U_3 is similar, where now $U_3(q[i]) = \max_s(\min_j(\max(u[i, s, j], U_2(q[j]))))$. The algorithm is repeated until U_N is obtained. The values $U_N(q[i])$ of this function represent the minimum control bound necessary to force the orbit starting in $q[i]$ to escape from Q within N or less iterations.

In order to define the algorithm to compute the escape functions, we define $u_{out}[i, s]$ as the control applied to the image $f(q[i], \xi[s])$ to move it outside the region Q. Then, U_1 is calculated as follows.

$$U_1(q[i]) = \max_s(u_{out}[i, s]) \tag{3}$$

If $q[i] + \xi[s]$ is beyond Q, then $u_{out}[i, s] = 0$. Then, given U_1 as the seed function, we can calculate the next escape functions with the following recursive algorithm

$$U_{k+1}^{in}(q[i]) = \max_s \left(\min_j \left(\max(u[i, s, j], U_k(q[j])) \right) \right) \tag{4}$$

$$U_{k+1}(q[i]) = \min \left(U_1(q[i]), U_{k+1}^{in}(q[i]) \right), \tag{5}$$

where the intermediate function U_{k+1}^{in} was introduced to allow the orbit to escape from Q before N iterations. Notice that U_{k+1}^{in} only takes into account images inside Q to control the orbit, while U_1 only takes into account images outside Q. Between these two possibilities, the one that minimizes the control will be chosen. By doing so, the orbit can be expelled in any of the $k \leq N$ iterations.

As an example, we have chosen $N = 3$ so that controlled orbits will escape from Q in 3 or less iterations. The upper disturbance bound affecting the logistic map was set to $\xi_0 = 0.030$. The corresponding escape functions U_k are shown in Fig. 4, which as it can be observed take zero values in some intervals. This means that points in these intervals where $U_k = 0$ will escape from Q within k iterations for any $\xi[s]$, without applying any control. It can be also observed that as the index k increases, the escape function decreases since the orbit has more iterations to escape.

Once we compute the escape functions, we have to select the control value u_0 to compute the corresponding escape sets E_k. No tall u_0 values are allowed. Escape sets only exist for values $u_0 \geq min(U_N)$. The bigger u_0, the bigger the escape sets. These sets consist of points $q[i]$ satisfying the condition $U_k(q[i]) \leq u_0$. The escape sets E_k for $u_0 = 0.022$ are represented in Fig. 4. For this case, the set E_3 represents the set of points $q[i]$ that can escape from Q within 3 iterations of the map,

by applying a control $u_n \leq u_0$ at each iteration. In the first iteration, the control u_1 will be applied to put the orbit in the closest point of E_2 or outside Q if possible. In the second iteration, the control u_2 will be applied to put the orbit in the closest point of E_1 or outside Q if possible. In the third and last iteration the control u_3 will be applied to put the orbit outside Q. This is illustrated in Fig. 5.

We have built in Fig. 6 a colormap plot showing the minimum number of iterations N needed for every orbit to escape from $Q = [0,1]$ in a (ξ_0, u_0) parameter plane. In this figure the white points correspond to the case where some initial conditions cannot escape from Q using these values of (ξ_0, u_0). A red line is plotted representing the ratio $u_0 = \xi_0$, so that values below this line are the ones we are interested due to the partial control method. Furthermore, we have chosen two points marked in red. The (+) point corresponds to $N = 4$, while the (*) point to $N = 19$.

3.2 Case B: Escape exactly in N iterations

Here we analyze the case when the number of iterations for an orbit to escape from Q is exactly N. A relevant observation here is that we need to control the orbit inside Q for $N - 1$ iterations, since it will escape precisely at the iteration N. The algorithm to obtain the escape functions is now simpler, because we do not need to consider the possibility that the orbit abandons Q before the iteration N, and is described next

$$U_1(q[i]) = \max_s(u_{out}[i,s]) \tag{6}$$

$$U_k(q[i]) = \max_s \left(\min_j \left(\max(u[i,s,j], U_k(q[j])) \right) \right). \tag{7}$$

To show how to control orbits to escape from Q in exactly 3 iterations, we use the logistic map with $\mu = 4.7$, and $\xi_0 = 0.030$. The escape functions are represented in Fig. 7. As it can be observed, the escape functions in case B have bigger values than in case A. The reason could be that the condition "expel the orbit in exactly N iterations" (case B) is stronger than the condition "expelling the orbit in N or less iterations" (case A). Therefore, we need bigger controls for the case B than the case A.

To get the escape sets E_k, we have chosen the control bound $u_0 = 0.022$. These sets are represented in Fig. 7. An scheme of how an orbit starting in E_3 escapes $Q = [0,1]$ in exactly 3 iterations is shown in Fig. 8.
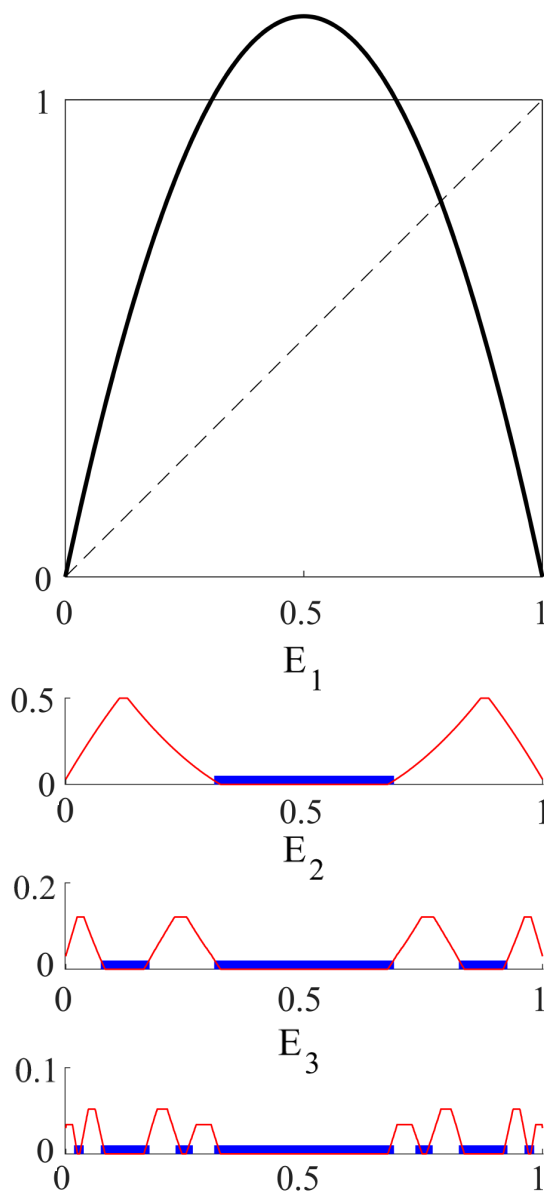

Fig. 4 Case A. On top, the logistic map with $\mu = 4.7$. On the bottom, the escape functions U_k are computed for $N = 3$ and shown in red. The escape sets E_k are shown in blue. We have used $\xi_0 = 0.030$ and $u_0 = 0.022$. Controlled orbits starting in E_3 will escape from $Q = [0,1]$ in 3 or less iterations by applying a control $|u_n| \leq u_0$ at each iteration. The scales used in the vertical axis are different for a better visualization.

4 Case C: Alternating chaotic regions

In the previous section, we mentioned three different cases to explore the escaping of an orbit from Q in N iterations. We discussed earlier, cases A and B. Now, we focus our attention on the third case corresponding to multistable chaotic systems that merge into a
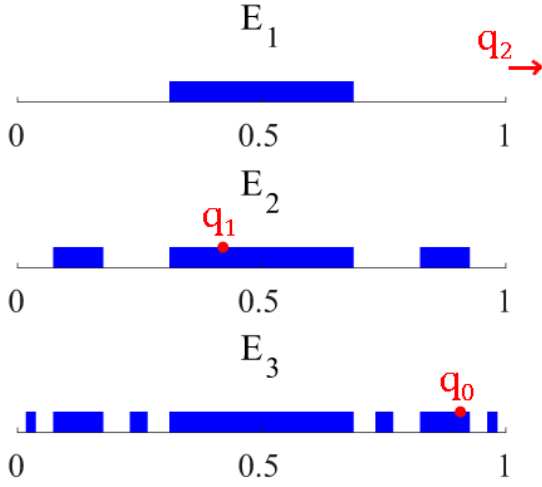
Fig. 5 Case A. Example of a controlled orbit. An initial condition in E_3 is controlled to E_2, then E_1, and finally it leaves the chaotic region $Q = [0, 1]$ under a suitable control. In general, orbits can escape in 3 or less iterations depending on the initial condition and the disturbance ξ_n. At each iteration, the applied control u_n is the minimum between the nearest escape set and the end points of Q. We have fixed here $\mu = 4.7$, $\xi_0 = 0.030$ and $u_0 = 0.022$
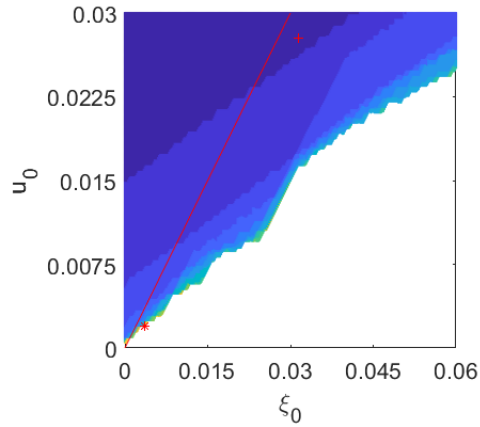

Fig. 6 This color plot shows the minimum number of iterations N needed for every orbit to escape from $Q = [0, 1]$ for any given couple of parameters (ξ_0, u_0), while the white points correspond to a situation where some initial conditions cannot escape. The red line represents the ratio $u_0 = \xi_0$. As an illustration we have marked two points in red. The (+) point corresponds to $N = 4$, while the (*) point to $N = 19$.

larger chaotic attractor as a parameter is varied [11–13]. This process occurs mainly when the basin boundary of each attractor collide, so that an orbit moves chaotically back and forth from one region to the other [14].

Our goal here is to use the control algorithm to force the orbit to stay in each region for a fixed number of iterations before moving to the other region. What we want here is to maintain a perpetual periodical motion
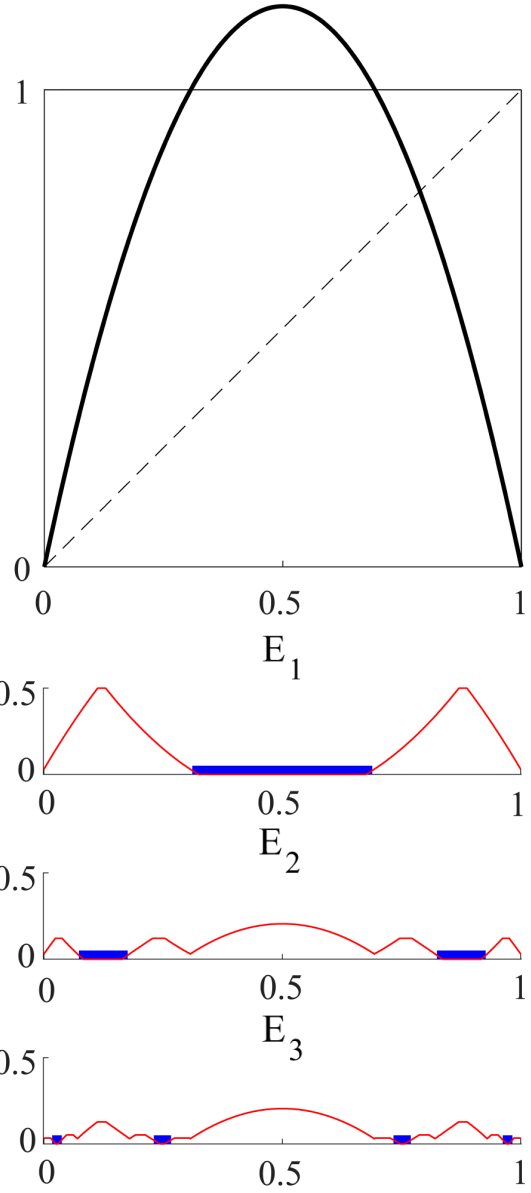

Fig. 7 Case B. On top, the logistic map with $\mu = 4.7$. On the bottom, the escape functions U_k are computed for $N = 3$ and shown in red. The escape sets E_k are shown in blue. We have used $\xi_0 = 0.030$ and $u_0 = 0.022$. Orbits starting in E_3 will escape from $Q = [0, 1]$ in exactly 3 iterations with a control $|u_n| \le u_0$ at each iteration.

bouncing back and forth between regions. The initial chaotic orbit stays for N iterations in one region, then it moves to the second region where it stays for N' iterations, and finally it comes back again to the first region. As we will see, the orbit will resemble a chaotic signal modulated by a periodic one.
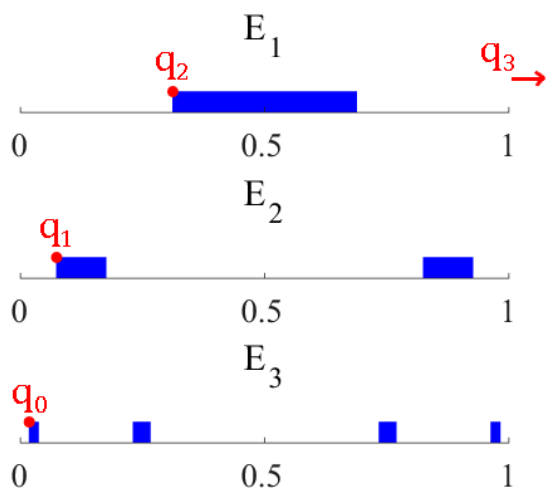
Fig. 8 Case B. Example of a controlled orbit. An initial condition in E_3 is mapped to E_2, then E_1, and finally it leaves the chaotic region $Q = [0,1]$ after a suitable control. All orbits starting in E_3 will escape $Q = [0,1]$ in exactly 3 iterations.

We have constructed a map that we name *the double parabola map* illustrated in Fig. 9 as an example of a simple map exhibiting the behavior described before,

$$x_{n+1} = \begin{cases} -\mu(x_n^2 + \frac{1}{2}x_n) & \text{si } x < 0.5, \\ 1 + \mu(x_n^2 - \frac{3}{2}x_n + \frac{1}{2}) & \text{si } x_n \geq 0.5, \end{cases} \tag{8}$$

This map is defined by a convex parabola at the region of the left $Q_l = [0, 0.5)$ and a concave parabola at the right region $Q_r = [0.5, 1]$. Both parabolas depend on the same parameter μ.

To show an example of the application of our logarithm to this case, we will focus on the behavior of the map for $\mu = 10$, where we have a chaotic attractor that expands to all the interval $[0, 1]$. Thus, orbits starting in the region Q_l behave chaotically to eventually escaping to the region Q_r, and vice versa as shown in the Fig. 9. Even though the map is well defined to map points of $[0, 1]$ into itself, however this could not be so in presence of a disturbance. In particular, for points close to the end of interval, what it should be considered in the control scheme.

The main purpose now is to apply a control so that the transition from the two regions would be predictable. In other words, we want to control how many iterations a given orbit stays on each region. As a consequence, our goal here is to keep the orbit N_l iterations on region Q_l and N_r iterations on region Q_r, where the values N_l and N_r are previously chosen by the controller. As a result, we will get a chaotic orbit that periodically oscillates between regions Q_l and Q_r.
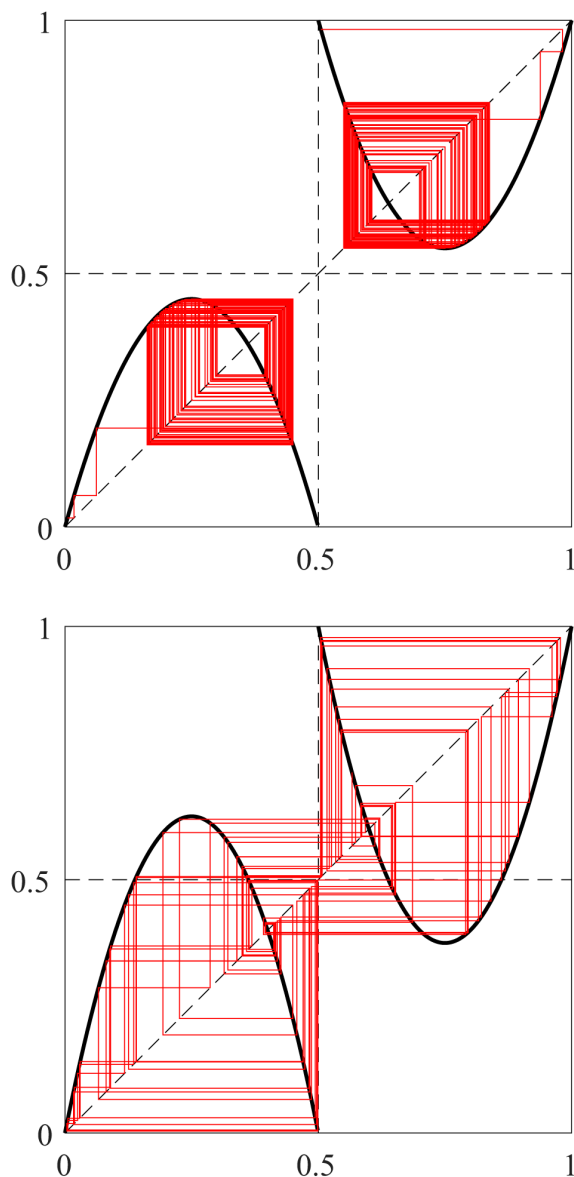

Fig. 9 Double parabola map. The map is defined in a piecewise manner in two different regions $Q_l = [0, 0.5)$ and $Q_r = [0.5, 1]$. (Top) When $\mu = 7.2$, the orbits stay on one side of the map, so there are clearly two different attractors. (Bottom) When $\mu = 10$, a global chaotic attractor merges. Now, orbits starting in Q_l have a transient chaotic behavior before escaping to the region Q_r where after another chaotic transient the orbit comes back to Q_l.

To compute the escape functions, we follow a similar methodology as the one used on case B from the previous section, though we must impose in the algorithm the periodic condition. Here, we will have N_l escape functions for orbits in Q_l and N_r escape functions for orbits in Q_r. From now on, we will denote as U^l and E^l the escape functions and escape sets of the left region

Q_l. Similarly U^r and E^r are defined in the right region Q_r.

Next, we briefly describe the algorithm to compute the escape functions. First, we start by computing the function

$$U_1^l(q[i]) = \max_s \left(\min_j \left(u_{in}^l[i, s, j] \right) \right). \qquad (9)$$

Taking this function as a seed, we can compute the rest of the escape functions with the following algorithm

Loop until the functions U_k^l and U_k^l converges
$\quad$ **for** $k = 1 : N_l - 1$ **do**
$\quad\quad$ $U_{k+1}^l(q[i]) =$
$\quad\quad$ $\max_s \left(\min_j \left(\max(u_{in}^l[i, s, j], U_k^l(q[j])) \right) \right)$
$\quad$ **end**
$\quad$ $U_1^r(q[i]) =$
$\quad$ $\max_s \left(\min_j \left(\max(u_{out}^r[i, s, j], U_{N_l}^l(q[j])) \right) \right)$
$\quad$ **for** $k = 1 : N_r - 1$ **do**
$\quad\quad$ $U_{k+1}^r(q[i]) =$
$\quad\quad$ $\max_s \left(\min_j \left(\max(u_{in}^r[i, s, j], U_k^r(q[j])) \right) \right)$
$\quad$ **end**
$\quad$ $U_1^l(q[i]) =$
$\quad$ $\max_s \left(\min_j \left(\max(u_{out}^l[i, s, j], U_{N_r}^r(q[j])) \right) \right)$

where u_{in}^l denotes the control applied to remain in Q_l and u_{in}^r is the one to remain in Q_r. On the other hand, u_{out}^r denotes the control needed to migrate from Q_r to Q_l and conversely u_{out}^l is the control to migrate from Q_l to Q_r.

Now, we want to illustrate the computation of a simple example where we have chosen $N_l = 2$, $N_r = 3$ and the upper disturbance bound $\xi_0 = 0.015$. The computed escape functions U^l and U^r are shown in Fig. 10. For the computation of the escape sets E_k, we choose the control bound $u_0 = 0.014$, which corresponds to the minimum value of the escape functions, and compute the points satisfying the conditions $U_k^l \leq u_0$ and $U_k^r \leq u_0$ so that we obtain E_k^l and E_k^r, respectively, as shown in Fig. 10. To control the orbit, we need to choose an initial condition in one of the escape sets E_k^l or E_k^r, and then a suitable control is applied. For example, if we start with a point in E_3^r, we apply the control in the next iterations to put the orbit in $E_2^r \rightarrow E_1^r \rightarrow E_2^l \rightarrow E_1^l \rightarrow E_3^r \rightarrow E_2^r \rightarrow ...(repeat)$. As a result, we obtain a chaotic motion modulated by a periodic one.
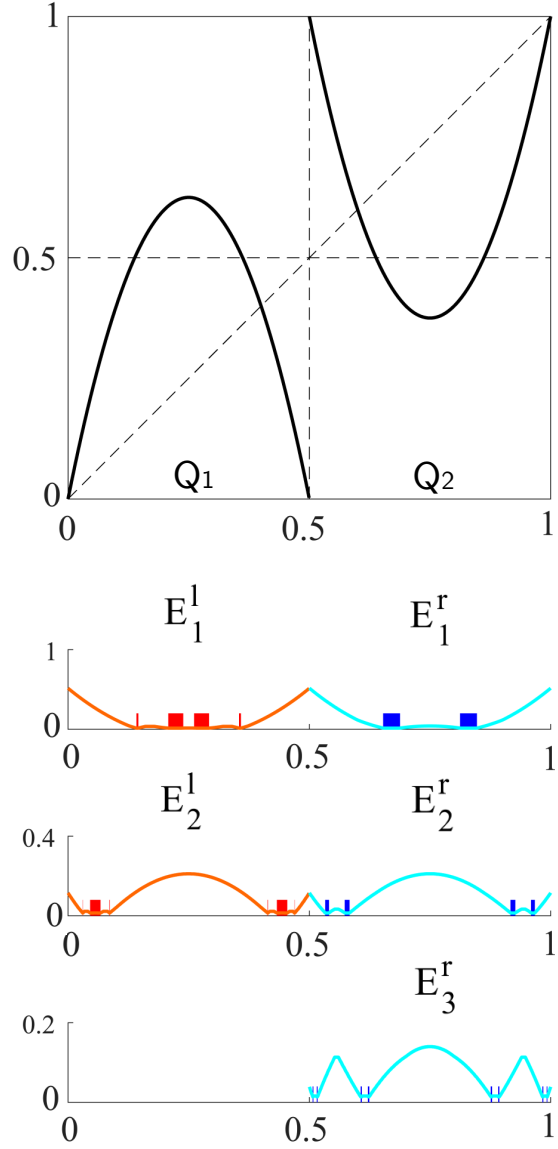

Fig. 10 On the top, the double parabola map with $\mu = 10$. On the bottom, the escape functions computed for $N_l = 2$ (orange) and $N_r = 3$ (cyan). Furthermore, the escape sets E^r appear as blue boxes and E^l as red boxes. We have used $\xi_0 = 0.015$ as the disturbance bound and $u_0 = 0.014$ as the control bound, where this one corresponds to the minimum value of the escape functions.

To make this behavior even more clear, we show another example considering now $N^l = 20$, $N^r = 30$, $\xi_0 = 0.015$ and control bound $u_0 = 0.0135$. This is shown in Fig. 11, on top it appears a typical orbit of the map and in the middle the controlled orbit remaining 20 iterations in $Q_l = [0, 0.5)$ and 30 iterations in $Q_r =$
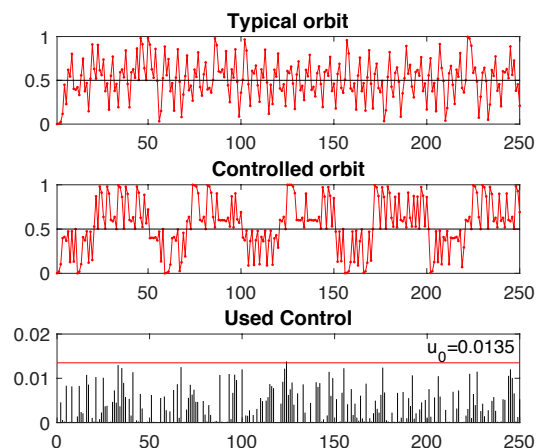
Fig. 11 On the top, a typical orbit for the double parabola map ($\mu = 10$). In the middle, we show an orbit affected by a disturbance $\xi_0 = 0.0150$ where we have used an upper control bound $u_0 = 0.0135$ to keep the orbit $N^l = 20$ iterations in $Q_l = [0, 0.5]$ and $N^r = 30$ iterations in $Q_r = [0.5, 1]$. At the bottom, the absolute value of the control used during the 250 iterations is represented. Note that all the applied controls remain below the control bound $u_0 = 0.0135$ shown by the red line, which corresponds to the minimum value of the escape functions.

$[0.5, 1)$. On the bottom, the values of the applied control are shown, which are all below $u_0 = 0.0135$.

5 Conclusions

We have developed a new control algorithm based on the partial control method aiming at keeping an orbit on a certain region Q for a given number of iterations N with a minimum control. For that purpose, we have adapted known tools such as safe functions and safe sets to the new escape functions and the escape sets. Once the latter are computed, it is straightforward to control the orbit.

We have considered three different possible scenarios. The first case, case A, where the main goal has been to force the escape of a given orbit of the phase space Q in N or less iterations. The second case, case B, where the goal has been to force the escape of the orbit to happen in exactly N iterations.

And the third scenario, case C, where we use the algorithm in a situation where a system has chaotic transitions between two regions, with the goal to control these transitions. As a consequence, this allows the controller to fix the number of iterations that the orbit will stay in every particular region, making it to have a periodic sequence of transitions.

Even though we have used one-dimensional maps for simplicity, we believe our control method is valid for higher dimensions, and we hope that it can be applied to different problems modeled with maps.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest concerning the publication of this manuscript.

References

1. E. Ott, C.Grebogi, and J. A. Yorke *Controlling chaos*, Phys. Rev. Lett. **64**, 1196 (1990)
2. K. Pyragas *Continuous control of chaos by self-controlling feedback*, Phys. Lett. A **170**, 421–428 (1992)
3. S. Zambrano, M. A. F. Sanjuán, and J. Yorke *Partial control of chaotic systems*, Phys Rev E **77**, 055201(R) (2008)
4. S. Zambrano, and M. A. F. Sanjuán *Exploring partial control of chaotic systems*, Phys Rev E **79**, 026217 (2009)
5. J. Sabuco, S. Zambrano, M. A. F. Sanjuán, and J. Yorke *Finding safety in partially controllable chaotic systems*, Commun Nonlinear Sci Numer Simulat **17** 4274-4280 (2012)
6. J. Sabuco, M. A. F. Sanjuán, and J. Yorke *Dynamics of partial control*, Chaos **22**, 047507 (2012)
7. R. Capeáns, J. Sabuco, and M. A. F. Sanjuán, *When less is more: Partial control to avoid extinction of predators in an ecological model*, Ecol. Complex. **19**, 1–8 (2014)
8. R. Capeáns, J. Sabuco, and M. A. F. Sanjuán, *Escaping from a chaotic saddle in the presence of noise*, Int. J. Dyn. Control. **1**, 78-86 (2018)
9. R. Capeáns, J. Sabuco, and M. A. F. Sanjuán, *Partial control of chaos: How to avoid undesirable behaviors with small controls in presence of noise*, Discrete Cont Dyn-B **23**, 3237–3274 (2018)
10. R. Capeáns, J. Sabuco, and M. A. F. Sanjuán, *A new approach of the partial control method in chaotic systems*, Nonlinear Dyn. **98**, 873–887 (2019)
11. E. L. Rempel and A. C.-L. Chian, *Intermittency induced by attractor-merging crisis in the kuramoto-sivashinsky equation*, Phys Rev E **71**, 016203 (2005)
12. A. L. Livorati, I. L. Caldas, C. P. Dettmann, and E. D. Leonel, *Crises in a dissipative bouncing ball model*, Phys Lett A **379**, 2830–2838 (2015)
13. S. Vaidyanathan, S. T. Kingni, A. Sambas, M. A. Mohamed, and M. Mamat, *A new chaotic jerk system with three nonlinearities and synchronization via adaptive backstepping control*, Int. J. Eng. Technol. **7**, 1936–1943 (2018)
14. Y. F. Jin, *Stochastic resonance in an under-damped bistable system driven by harmonic mixing signal*, Chinese Phys B **27**, 050501 (2018)